Journal of Information Technology and Applications

(BANJA LUKA)

Exchange of Information and Knowledge in Research

VOLUME 1 NUMBER 2 BANJA LUKA, DECEMBER 2011 (77-148) ISSN 2232-9625 (Print) UDC 004

APEIRON

THE AIM AND SCOPE

The aim and scope of the Journal of Information Technology and Applications (JITA) is:

- to provide international dissemination of contributions in field of Information Technology,
- to promote exchange of information and knowledge in research work and
- to explore the new developments and inventions related to the use of Information Technology towards the structuring of an Information Society.

JITA provides a medium for exchanging research results and achievements accomplished by the scientific community from academia and industry.

By the decision of the Ministry of Education and Culture of the Republic of Srpska, no.: 07.030-053-160-4/10 from 3/3/2010, the journal „Journal of Information Technology and Applications“ Banja Luka is registered in the Registry of public organs under the number 591.
Printed by Markos, Banja Luka in 300 copies two times a year.

UDC 004

ISSN 2232-9625 (Print)
ISSN 2233-0194 (Online)

JITA

VOLUME 1 NUMBER 2 BANJA LUKA, DECEMBER 2011 (77-148)

Contents

Editorial

The content of the second issue of JITA consists of seven papers. The first paper, entitled "Functional dependencies analysis in fuzzy relational database models," by Miljan Vučetić, presents a literature overview of Fuzzy Relational Database Models with emphasis on the role of functional dependencies in logical designing and modeling. Fuzzy set theory is powerful tool for manipulating imprecise and uncertain information and now is widely applied for the classical relational database extensions resulting in numerous contributions.

The next paper "Mutation Testing: Object-Oriented Mutation and Testing Tools," by Zdravko Ivanković, Branko Markoski, and Dragica Radosav, tackles the problem of software testing using mutation testing technique. The basic idea of mutation testing is to seed lots of artificial defects into the program, test all defects individually, focus on those mutations that are not detected, and, finally, improve the test suite until it finds all mutations. Procedure-oriented mutation systems make mutations of expressions, variables and statements, but do not mutate type and component declarations. Object-oriented programming features changed the requirements for mutation testing. Mutation testing requires automated testing tools, which is not a trivial tool to make.

The third article "Comparative Implementation Analysis of AES Algorithm" by Boris Damjanović, and Dejan Simić, describes the results of a study which compares performance of well known cryptographic packages, Oracle/Sun and Bouncy Castle implementations, in relation to author's small and specialized implementations of AES algorithm. The evaluation results show that Bouncy Castle and Oracle/SUN gave pretty equal performance results. Proposed novel implementation of AES algorithm showed some advantages related not only to algorithm speed, but also to possibilities for further analysis of the algorithm.

In "The way of students' efficiency improvement in knowledge acquisition and transfer knowledge model in Clarolina CMS", Nevzudin Buzadjija presents results of the research in using Clarorina e-learning system, which was organized in one high school in Bosnia and Herzegovina. The research was conducted from the subject informatics in the I, II and III grade. The aim of this paper is to increase motivation of high school students with regard to the use of online materials.

"Digital Signal Processing Applications with Iterative Logarithmic Multipliers" by Aleksej Avramović, Patricio Bulic, and Zdenka Babić, discusses logarithm-based approximate multipliers and squarers, their characteristics and digital signal processing applications based on approximate multiplications. Their iterative multipliers and squarers contain arbitrary series of basic blocks that involves only adders and shifters enabling fast execution, less power-consuming and high accuracy of implemented algorithms. It was shown that this approach can be used in several signal processing applications without decreasing of application efficiency.

Dušan Okanović, Milan Vidaković, and Zora Konjović, in "Monitoring of JEE applications and performance prediction" present one solution for continuous monitoring of JEE application. This paper outlines the architecture and basic functionality of the Kieker framework and how it can be extended for adaptive monitoring of JEE applications. Collected data was used for analysis of application performance. In order to predict application performance, the regression analysis was employed.

The last article in this issue "Social Media in Marketing and PR" by Velimir Štavljanin, Vinka Filipović, and Milica Kostić Stanković, introduces the role of social media in contemporary marketing and PR. Social media as a new communication channel has managed to radicalize the way companies communicate with consumers and other stakeholders. Companies that are not on time engaged in social media weaken its abil-

ity for competitive struggle. This paper presents possibilities of different types of social media in relation to marketing and public relations. Also, specific recommendations for the use of social media in marketing and public relations are proposed.

On behalf of the Editorial Board we would like to thank the authors for their high-quality contributions, and also the reviewers for the effort and time invested into the preparation of this issue of Journal of Information Technology and Applications.

EDITORS: Gordana **Radić**, Editor-in-Chief, Zoran **Avramović**, Dušan **Starčević**

Digital Signal Processing Applications with Iterative Logarithmic Multipliers

[1]Aleksej Avramović, [2]Patricio Bulić, [3]Zdenka Babić

[1](aleksej@etfbl.net), [2](patricio.bulic@fri.uni-lj.si), [3](zdenka@etfbl.net)

Contribution to the State of the Art

UDC **621.391:004**

Abstract: Many digital signal processing applications demand a huge number of multiplications, which are time, power and area consuming. But input data is often corrupted with noise, which means that a few least significant bits do not carry usable information and do not need to be processed. Therefore, approximate multiplication does not affect application efficiency when approximation error is less than noise introduced during data acquisition. This fact enables usage of faster and less power-consuming algorithms that is important in many cases where processing includes convolution, integral transformations, distance computations etc. This paper discusses logarithm-based approximate multipliers and squarers, their characteristics and digital signal processing applications based on approximate multiplications. Our iterative multipliers and squarers contain arbitrary series of basic blocks that involves only adders and shifters; therefore, it is not power and time consuming and enables achieving arbitrary accuracy. It was shown that proposed approximate multipliers and squarers can be used in several signal processing applications without decreasing of application efficiency.

Keywords: Approximate multiplication, Digital signal processing

Introduction

Digital signal processing (DSP) applications often involve algorithms, which demand a huge number of multiplications, which can be time, power and area consuming. Multipliers often process a large amount of data corrupted with noise, which is unnecessary consumption of power and time. For example, many applications involve calculations of integral transformations, such as Fast Fourier Transform (FFT), Discrete Cosine Transform (DFT) and Discrete Wavelet Transform (DWT), after which quantization is applied, like in algorithms for compression [3], [20]. Similarly, frequency leakage, which is common during spectrum analysis, may lead to the estimation of harmonics with certain amount of error. In such applications, which involve error due to quantization or other quantization, sometimes it is more efficient to calculate multiplication results without least significant bits, instead of calculating full-precision results. In other DSP applications convolution or correlation between two signals has to be calculated. Calculation of correlation may involve a large number of multiplications, but it is important to notice that only the maximum or its approximation, of correlation is used; therefore approximated multiplication will not decrease application efficiency. The similar is with noise filtering and other applications that include convolution. Other applications that involve a significant number of multiplications are found in cryptography, object matching and recognition, video and image processing, etc. In applications where the speed of the calculation is more important than accuracy, truncated or logarithm multiplications seem to be suitable methods [6], [12], [18].

Integer, truncated and logarithmic multipliers

Integer multiplier is one of the simplest methods for computation of the product, but it requires n multiplication steps for two n-bits unsigned numbers [10]. Such an integer multiplication, where the least-significant bit of the multiplicator is examined,

is known as the radix-2 multiplication. Shorter time delay can be achieved by examining *k* lower bits of the multiplicand in each step. Usually, the radix-4 multiplication is used, where two least-significant bits of the multiplicand are examined. This kind of approach usually requires significant hardware resources. The well-known implementation of such a multiplier is an array multiplier, where *n*-2 *n*-bits carry-save adders and one *n*-bits carry-propagate adder is used to implement the *n*-bits array multiplier.

Truncated multipliers are extensively used in digital signal processing where the speed of the multiplication and the area- and power-consumptions are important [11], [18], [22]. However, as mentioned before, there are many applications in DSP where high accuracy is not important. By discarding some of the less significant bits, which can be corrupted with noise, multiplier is less hardware and time consuming. If it is necessary, simple compensation circuits can be applied to reduce the approximation error [6], [14], [15].

Logarithmic multiplication is an approximate multiplication technique that uses the fact that logarithm of the product is a sum of operand logarithms [6], [12], [14], [15]; therefore an operand conversion from integer number system into the logarithm number system (LNS) is used. In more detail, the multiplication of the two operands N_1 and N_2 is performed in three phases, calculating the operand logarithms, the addition of the operand logarithms and the calculation of the antilogarithm:

$$\log(N_1 N_2) = \log(N_1) + \log(N_2) \tag{1}$$

The main advantage of this principle is that multiplication is done by one summation, but approximation of logarithm and antilogarithm conversion introduces error. An iterative approximation of LNS multiplier can be derived from binary representation of a number:

$$N = 2^k \left(1 + \sum_{i=j}^{k-1} 2^{i-j} Z_i \right) = 2^k \left(1 + x \right) \tag{2}$$

where *k* is place of the most significant bit equals one, so called characteristic number, and *Z* is a bit value at the *i*th position. Because, computers work with binary number system, it is most appropriate to use 2 as logarithm basis, so we can derive:

$$\log_2(N) = \log_2 \left(2^k \left(1 + \sum_{i=j}^{k-1} 2^{i-j} Z_i \right) \right) = \log_2 \left(2^k \left(1 + x \right) \right) = k + \log_2 \left(1 + x \right) \tag{3}$$

Previous equation is a basis for Mitchells LNS multiplier approximation first time presented in [16]. Second term in (3) is discarded as an approximation error, but Mitchell also suggested correction term based on if-else logic. Later, several authors tried to simplify correction in various ways. Abed and Sifred [1], [2] derived correction equations with coefficients that are a power of two, reducing the error and keeping the simplicity of the solution. Among the many methods that use look-up tables for error correction in the MA algorithm, McLaren's method [15], which uses a look-up table with 64 correction coefficients calculated in dependence of the mantissas values, could be selected as one that has satisfactory accuracy and complexity. A recent approach for the MA error correction, reducing the number of bits with the value of '1' in mantissas by operand decomposition, was presented by Mahalingam and Rangantathan [14]. LNS multipliers can be generally divided into two categories, one based on methods that use lookup tables and interpolations, and the other based on Mitchell's algorithm (MA) [16], although there is a lookup-table approach in some of the MA-based methods [14].

ITERATIVE MULTIPLIER AND SQUARER

In [5], [6] and [8], algorithm of iterative logarithmic multiplier is presented and analyzed in detail. Iterative calculation of correction terms is one way to deal with LNS multiplier approximation explained in (3). This kind of approach introduces a simple pipelined basic block for calculation of first approximation. Basic block avoids if-else logic, thus significantly reducing the hardware resources. The same basic block can be used for error correction, which represents a significant advantage for simpler hardware implementation. Due to optimal pipelining, correction term calculation may begin before the first approximation is calculated, saving calculation time. The second advantage of iterative approach is

the fact that arbitrary number of correction blocks can be added, without increasing the time of delay. From (3), we can derive true value of a product:

$$\begin{aligned} P_{true} &= N_1 N_2 = 2^{k_1}\left(1+x_1\right)2^{k_2}\left(1+x_2\right) \\ &= 2^{k_1+k_2}\left(1+x_1+x_2\right)+2^{k_1+k_2}\left(x_1 x_2\right) \end{aligned} \quad (4)$$

Combining (4) with (1) it can be shown that:

$$P_{true} = 2^{k_1+k_2}+\left(N_1-2^{k_1}\right)2^{k_2}+\left(N_2-2^{k_2}\right)2^{k_1}+\left(N_1-2^{k_1}\right)\left(N_2-2^{k_2}\right) \quad (5)$$

We can see that the last term in (5) demands another multiplication, so by discarding it, we can introduce the first approximation:

$$P_{ap} = 2^{k_1+k_2}+\left(N_1-2^{k_1}\right)2^{k_2}+\left(N_2-2^{k_2}\right)2^{k_1} \quad (6)$$

which can be implemented easily. In [6], it was proven that adding of finite number of correction terms could reduce an approximation error arbitrary. In Figure 1, the pipelined version of iterative multiplier is shown, while Figure 2 depicts an iterative logarithmic multiplier with one correction circuits.

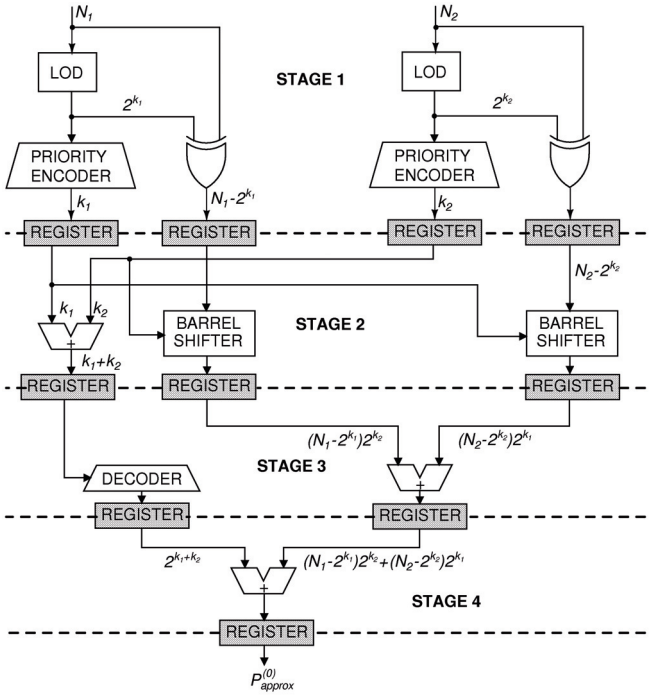


FIGURE 1. FOUR STAGE PIPELINED VERSION OF ITERATIVE LOGARITHMIC MULTIPLIER'S BASIC BLOCK.

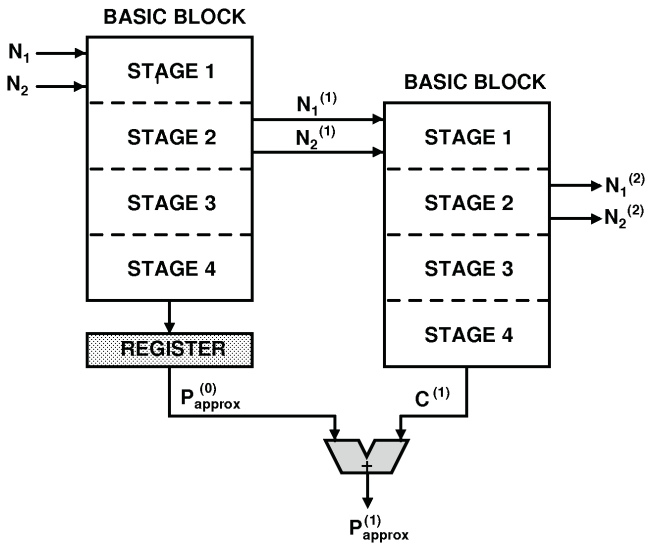


FIGURE 2. AN ITERATIVE LOGARITHMIC MULTIPLIER WITH BASIC BLOCK AND ONE ERROR CORRECTION CIRCUITS.

In many digital signal applications, for example, for Euclidean distance calculation, a large number of squaring is employed. For this purpose, an iterative multiplier can be used, but exploiting the fact that the operands are same, can lead to even further hardware simplification. In (7), a simple logarithmic squarer is described. Similar to the multiplier, an approximate equation can be derived for iterative squarer as well. Correct value of a square of N is:

$$\begin{aligned} S_{true} &= \left(2^k+\left(N-2^k\right)\right)\left(2^k+\left(N-2^k\right)\right) \\ &= 2^{2k}+\left(N-2^k\right)2^{k+1}+\left(N-2^k\right)^2 \end{aligned} \quad (7)$$

We can see that the last term in (7) demands another square, so by discarding it, we can introduce the first approximation of a square:

$$S_{ap} = 2^{2k}+\left(N-2^k\right)2^{k+1} \quad (8)$$

Similarly to logarithmic multiplier, in [19] it was proven that finite number of correction circuits could lead to arbitrary small approximation error. The first approximation of square, given by (8), requires one logical shift left (no gates required), one subtraction and one shift by k (Barrel shifter required).

MOTION VECTOR DETECTION

Motion vector is widely used in video compression applications and standards, such as MPEG [23],

as well as for moving object location and tracking [7], [21]. In [6] the use of iterative logarithmic multiplier for motion vector detection is described. Direct and most accurate method for motion vector detection is based on technique of block matching, which requires computation of block correlation. The larger is the block, the more multiplications must be calculated for correlation computing. For efficient compression, a compromise between the speed of the calculation and the accuracy of the motion vector is necessary.

We considered matching techniques based on a block correlation. If we take two successive or near video frames and mark them as the reference frame and the observed frame, motion vector technique tries to match blocks from reference frame and observed frame. It is important to find a matching for each block from observed frame (observed block). Motion vector is used as a measure of distance between same object in reference and observed frame. Usually, the difference between successive or near-successive video frames is very small, thus coding that difference may result in faster and more efficient compression. In moving object location and tracking we try to find motion vectors that belong to many objects. If we denote the observed block with *F(i,j)*, where *i* and *j* are the pixels' coordinates, and a respective block in the reference frame with *S(i,j)*, assuming the block size is *N*x*N*, then the correlation coefficients *C(x,y)* are calculated for all positions *(x,y)* from the reference region as follows:

$$C(x,y)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}F(i,j)S(x+i,y+j) \qquad (9)$$

As we can see from equation (9), for each pixel in the block, *N*x*N* multiplications must be calculated, which means, that for the block size is *N*x*N*, N^4 multiplications must be calculated. In cases where large frame video stream is processed, this number can be enormous. Hence, other nonlinear methods for block matching can be used, which can introduce matching error [23]. As we can see, calculation of correlation between to blocks can be very computationally expensive, but it is important to notice that only the position of correlation maximum is required for estimation of motion vector. Often we are satisfied with near maximum position, which leads to near accurate motion vector estimation, and will not decrease algorithm efficiency significantly. If approximate multiplication is used instead of full-precision multiplication, most of the correlation coefficients will be decreased for certain percentage. Maximum of correlation function usually will not be different comparing to algorithm with full-precision multipliers. In [6] it was shown that correlation-based block matching technique with iterative logarithmic multiplier and only one error correction circuit introduces mismatch percentage about 3%, while iterative logarithmic multiplier with two correction circuits has negligible mismatch percentage. Mismatch is defined as a difference of maxima of correlation function compared with application with full-precision multipliers. Therefore, approximate multipliers with less power and time consumption can be used in this application.

SYSTEM IDENTIFICATION

System identification process tries to describe unknown system with linear and time invariant model, which has the same behavioral characteristics as observed unknown system. System identification is usually done with some kind of adaptive filter. Coefficients of model are adapted until the difference between these two systems output becomes arbitrary small. One of the most popular methods for adaptation is based on minimization of mean square difference, and it is called Least Mean Square (LMS) algorithm. LMS algorithm can require a significant number of multiplications, especially for high order system and system with slow convergence. In [4] the fixed-point LMS algorithm, based on iterative logarithmic multiplier is described and tested. If we use $\mathbf{h}(n)$ to denote adaptive filter coefficients vector after *n*-th step, we can derive equation for coefficients correction in next step:

$$\mathbf{h}(n+1)=\mathbf{h}(n)-\frac{\mu}{2}\nabla J(n) \qquad (10)$$

where μ is adaptation step unit and *J(n)* is error cost function for previous step. Error cost function estimates the difference between responses of the unknown system and the model. If we use mean square error for the difference measure, cost function can be estimated as a mathematical expectation

of product of adaptive filter response and adaptation error, which yields:

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mu E\left(\mathbf{x}(n)e(n)\right) \quad (11)$$

where $\boldsymbol{x}(n)$ denotes adaptive filter response and $e(n)$ denotes error in the n-th step. Estimation of mathematical expectation E depends on number of samples and can demand a large number of multiplications, especially if the system order is high. LMS algorithms adapted for fixed-point systems have several rounding error types, for example, input data rounding error, coefficients rounding error, etc. In such algorithm realizations, adaptation time can be decreased with approximate multipliers. An approximation error can be treated as one of the rounding errors. In [4] it was shown that approximation error can be treated as one of rounding errors and that even the logarithmic multiplier with basic block only, would not affect adaptation convergence.

Context-based image retrieval

It is often necessary to find contextually similar images with query-image, from large number image datasets. Recently, image datasets can consist of several thousands to several hundred thousands images, therefore, searching for visually similar images is likely impossible. Context-based image retrieval (CBIR) system describes every image with appropriate descriptor that is associated to each image. Descriptor has various types of information about image properties, like low level description of color, shapes and textures, and higher-level structures like context. Query is done by calculating distance between query-descriptor and every descriptor from dataset. Descriptors can be relatively high dimensionality, so calculating Euclidean distance between every mage descriptor from database can demand a huge number of squaring.

Logarithmic squaring is a simpler version of logarithmic multiplier; therefore it requires less time and power. In [19] CBIR system based on logarithmic squarer is described, and it was proven that system efficiency is not compromised. Images were represented using Gist descriptor, as it was described in [9] and [17]. Gist descriptor tries to describe image at local spatial level. For color images, descriptor dimensionality can be more that 1500, so calculation of Euclidean distance between two images may demand more than 1500 squaring. Modern digital image databases contain more than several hundred thousand images, so it is obvious that squaring represent time bottleneck for large databases. On the other hand, a large dimensionality descriptors are often corrupted with noise, thus approximate squarers probably will not decrease efficiency of CBIR system. In [19] mean average precision (MAP), of CBIR system with full-precision and approximate logarithmic squarers are compared. Seventy queries were performed to find near duplicate images on dataset contained of 10 000 images. Original images were not considered as neither correct nor incorrect retrievals. It was shown that MAP of system with approximate logarithmic squarer with one error correction circuit has same value as MAP of system with full-precision multiplier. Therefore, it was proven that approximate squarer can be used for image retrieval efficiently.

Neural network application

The hardware implementations of artificial neural network models have found their place in some niche applications like image processing, pattern recognition, speech synthesis and analysis, adaptive sensors with teach-in ability and so on.

Neural networks offer a high degree of internal parallelism, which can be efficiently used in custom design chips. Neural network processing comprises of a huge number of multiplications, i.e. arithmetic operations consuming a lot of space, time and power. In [13] we have shown that exact matrix multipliers can be replaced with approximate iterative logarithmic multipliers with one error correction circuit. As neural networks have highly adaptive nature, which compensated the erroneous calculation, the replacement of the multipliers does not have any notable impact on the NN processing and learning accuracy. Authors in [13] proposed hardware implementation of the multilayer perceptron with on chip learning ability, which confirmed the potential of the proposed approximate multiplier. Authors in [13] performed experiments on Proben1 benchmark dataset,

which showed that the adaptive nature of the proposed neural network model enables the compensation of the errors caused by inexact calculations. The iterative logarithmic multipliers require less resource on a chip, which leads to smaller designs on one hand and on the other hand to designs with more concurrent units on the same chip. A consumption of fewer resources per multiplier also results in more power efficient circuits. In [13] we achieved about 20 % of the reduction in power consumption.

CONCLUSION

In this paper, possibilities of usage of approximate logarithmic multiplications and squaring, as a special case of multiplying, in various digital signal applications were described. We proposed logarithmic multipliers which belong to a class of approximate multipliers based on a trade-off principle. Trade-off between accuracy and low time and power consuming is performed in algorithms where low consumption is more important than accuracy. In this paper, several such algorithms are described. Examples of use of iterative logarithmic multipliers in various applications, such as motion vector detection, system identification, image retrieval and neural networks were presented. It was shown that approximation of multiplication does not affect digital signal processing application efficiency, especially when application estimations deal with noise-corrupted data.

REFERENCES:

[1] Abed, K.H. and Sifred, R.E. (2003). CMOS VLSI Implementation of a Low-Power Logarithmic Converter, IEEE Transactions on Computers, vol. 52, no. 11, pp. 1421-1433.

[2] Abed, K.H. and Sifred, R.E. (2003). VLSI Implementation of a Low-Power Antilogarithmic Converter, IEEE Transactions on Computers, vol. 52, no. 9, pp. 1221-1228.

[3] Agostini, L.V., Silva, I.S. and Bampi, S. (2007). Multiplierless and fully pipelined JPEG compression soft IP targeting FPGAs, Microprocessors and Microsystems, vol. 31, issue 8, pp. 487–497.

[4] Avramović, A., Risojević, V., Babić, Z. and Bulić, P. (2010). System Identification Using Least Mean Square Algorithm with Logarithmic Multiplier, In Proceedings of 8th Symposium INDEL, Banja Luka, BIH, pp. 134-137.

[5] Babić, Z., Avramović, A. and Bulić, P. (2008). An Iterative Mitchell's Algorithm Based Multiplier, In Proceedings of The IEEE Symposium on Signal Processing and Information Technology, Sarajevo, BIH, pp. 303-308.

[6] Babić, Z., Avramović, A. and Bulić, P. (2011). An Iterative Logarithmic Multiplier, Microprocessors and Microsystems, vol. 35, issue 1, pp. 23-33.

[7] Babić, Z., Ljubojević, M. and Risojević, V. (2011). Indoor RFID Localization Improved by Motion Segmentation, In Proc. 7th International Symposium on Image and Signal Processing and Analysis, pp. 271-276.

[8] Bulić, P., Babić, Z. and Avramović, A. (2010). A Simple Pipelined Logarithmic Multiplier, In Proceedings of 28th International Conference on Computer Design ICCD, Amsterdam, Netherlands, pp. 235-240.

[9] Douze, M., Jegou, H., Sandhawalia, H., Amsaleg, L. and Schmid, C. (2009). Evaluation of gist descriptors for web-scale image search, in International Conference on Image and Video Retrieval, ACM.

[10] Hennessy, J.L. and Patterson, D.A. (2007). Computer Architecture: A Quantitative Approach, fourth ed., Morgan Kauffman Pub.

[11] Kidambi, S.S., El-Guibaly, F. and Antoniou, A. (1996). Area-efficient multipliers for digital signal processing applications, IEEE Transactions Circuits and Systems II: Analog and Digital Signal Processing, vol. 43, no. 2, pp. 90–95.

[12] Kong, M.Y., Langlois, J.M.P. and Al-Khalili, D. (2008). Efficient FPGA implementation of complex multipliers using the logarithmic number system, In IEEE International Symposium on Circuits and Systems, ISCAS, pp. 3154–3157.

[13] Lotrič, U. and Bulić, P. (2011). Logarithmic multiplier in hardware implementation of neural networks, in: A. Dobnikar, U. Lotric, B. Ster (Eds.), ICANNGA (1), volume 6593 of Lecture Notes in Computer Science, Springer, pp. 158–168.

[14] Mahalingam, V. and Rangantathan, N. (2006). Improving Accuracy in Mitchell's Logarithmic Multiplication Using Operand Decomposition, IEEE Transactions on Computers, vol. 55, no. 2, pp. 1523-1535.

[15] McLaren, D.J. (2003). Improved Mitchell-based logarithmic multiplier for low-power DSP applications, In Proceedings of IEEE International SOC Conference, pp. 53-56.

[16] Mitchell, J.N. (1962). Computer multiplication and division using binary logarithms, IRE Transactions on Electronic Computers, pp. 512–517.

[17] Oliva, A. and Torralba, A. (2001). Modeling the shape of the scene: a holistic representation of the spatial envelope, International Journal of Computer Vision, vol. 42, no. 3, pp. 145–175.

[18] Rais, M.H. (2009). Efficient hardware realization of truncated multipliers using FPGA, International Journal of Applied Science, vol. 5, no. 2, pp. 124–128.

[19] Risojević, V., Avramović, A., Babić, Z. and Bulić, P. (2011). A Simple Pipelined Squaring Circuit for DSP, In Proceedings of 29th International Conference on Computer Design ICCD, Amherst, MA, USA, pp. 162-167.

[20] Srot, S. and Zemva, A. (2007). Design and implementation of the JPEG algorithm in integrated circuit, Electrotechnical Review, vol. 74, no. 4, pp. 165–170.

[21] Tekalp, A. M. (1995). Digital Video Processing, Prentice Hall.

[22] Van, L.-D. and Yang, C.-C. (2005). Generalized low-error area-efficient fixed-width multipliers, IEEE Transactions Circuits and Systems I: Regular Paper, vol. 52, no. 8, pp. 1608–1619.

[23] Watkinson, J. (2004). The MPEG Handbook: MPEG-1, MPEG-2, MPEG-4, second ed., Focal Press.

Submitted: December16, 2011
Accepted: December 31, 2011

FUNCTIONAL DEPENDENCIES ANALYSE IN FUZZY RELATIONAL DATABASE MODELS

Miljan Vučetić

Faculty of Organizational Science University of Belgrade

Contribution to the State of the Art

UDC **004.651**

Abstract: This paper presents a literature overview of Fuzzy Relational Database Models with emphasis on the role of functional dependencies in logical designing and modeling. The aim is the analysis of recent results in this field. Fuzzy set theory is widely applied for the classical relational database extensions resulting in numerous contributions. This is because fuzzy sets and fuzzy logic are powerful tool for manilupating imprecise and uncertain information. A significant body of research in efficient designing FRDM has been developed over the last decades. Knowing the set of functional dependencies, database managers have a chance to normalize the same eliminating redundancy and data anomalies. In this paper we have considered the most important results in this field.

Key words: fuzzy relational database model, functional dependencies, fuzzy functional dependencies, fuzzy set.

INTRODUCTION

Classical database models often suffer from their incapability of representing and manipulating with imprecise and uncertain information that appear in many real world applications. Since the early '80s, Zadeh's fuzzy logic has been used to extend different data models. The purpose of introducing fuzzy logic in the database is the possibility of representing and monitoring a vague and imprecise information. This resulted in numerous contributions, mainly in computer applications. Naturally, fuzzy relational database extends a function of classical data models, which provides a higher level of fuzzy system adaptation as one of the basic features of intelligent systems (in addition to system of planning, learning, prediction, system for knowledge search, robots). A very important thing of this data model is the fact that there are many active research areas that directly involve or use these knowledge base. The issue about vague and imprecise data and their representations is represented and important in various fields. We'll list just a few of them: geographic information systems (GIS) and representation of spatial data systems, data mining, statistical database models, information retrieval.

In a relational database models real interest is the identification of dependencies between data, i.e. functional and fuzzy functional dependencies, so that these models could be normalized. In this way, the database design is based on the assumption that there is a set of dependencies which is the input for database normalization. There have been a lot of papers about data dependency analyse, but there isn't comprehensive review in this area. In this paper, we have considered and systematically elaborated the concept of functional dependency that is extended to Fuzzy Relational Database Models (FRDM).

The remainder of this paper is organized as follows. Section 2. gives basic knowledge about fuzzy set theory and uncertain information. Fuzzy relational database models are described in section 3. Section 4. explores issues and papers in the field of functional dependencies analyze. The fifth section is scheduled for conclusion.

IMPERFECT INFORMATION AND FUZZY SET THEORY

Imprecise and unceratin information

Inconsistency, *imprecision*, *vagueness*, *uncertainty*, and *ambiguity* are five basic kinds of imperfect information in database systems [25].

a. Inconsistency is a kind of semantic conflict, meaning the same aspect of the real world is represented differently in one or in several different databases. For example, the *age* of one person is stored as 34 and 37 simultaneously.
b. Intuitively, the imprecision and vagueness are relevant to the content of an attribute value, which means that a atribute value must be made from a given range (interval or set) of values but we do not know exactly which value will be selected at present. In general, vague information is represented by linguistic variables. For example, the *young man* is a set {20,21,22,23} which means that a young man can be 20 or 23 years old.
c. The uncertainty is related to the degree of truth of its attribute value, and it means that we can apportion some, but not all, of our belief to a given value or a group of values. For example, the possibility that the *age* of *Marko* is 35 right now may be 97%. The random uncertainty described with probability theory is not considered here.
d. The ambiguity means that some elements of the data model lack complete semantics leading to several possible interpretations.

Generally, several different kinds of imperfection can co-exist regarding to the same data in database. For example, person's age is data from a set of values and their membership degrees are 0.85, 0.90, 0.96 and 0.80 respectively. Imprecision, uncertainty and vagueness are the most often types of imperfect information in classical relational database.

Fuzzy set theory and possibility distributions

Many of the existing approaches related to imprecision and uncertainty information are based on the theory of fuzzy sets and possibility distribution theory. A fuzzy set (0.85/20, 0.90/21, 0.96/22, 0.80/23) for the person's age contains uncertainty information (a person's age may be 20, 21, 22 or 23 years) and the degree of membership (0.85, 0.90, 0.96 and 0.80) simultaneously. One of the most important characteristics of fuzzy sets is their ability to express the degree of uncertainty in human thinking and his subjectivity. Such a basic idea with membership grade or weighted elements is proved as very useful in the knowledge analysis and information representation.

Let X be a domain. A fuzzy set A defined on X is usually displayed in the form:

$$\mu_A : A \rightarrow [0,1]$$

In this way, each element x in fuzzy set A has a degree of membership $\mu_A(x) \in [0,1]$. Thus the fuzzy set A is described as a set of n-tuples:

$$A = \{x, \mu_A(x) : x \in A\}$$

where $\mu_A(x)$ denotes the degree of membership of x in the fuzzy set A. When $\mu_A(x)$ is greater, there is more thruth in the claim that the element x belongs to A.

When X is an infinite set, fuzzy set A defined on X is represented as:

$$A = \int_x \frac{\mu_A(x)}{x}$$

Three major meanings for membership function which exist in the litareature are: similarity, preference and uncertainty. Each of these semantics can be used in real class of applications. Membership function of a fuzzy set is sometimes a kind of utility function that represents flexibile contstraints in the decision-making problems. In following paragraphs are defined interpretations of membership function using in applicatons.

Degree of similarity: membership function can be used for defining the degree of closeness and similarity between respective elements. This is also the oldest semantics introduced by Bellman et al. and this view is particularly significant in clustering

and regression analysis where we have considered the problem of data representing and determining closeness between them.

Degree of preference: a fuzzy set A represents a set of more or less preferred objects or values of the variables using for decision making. In this case, $\mu_A(x)$ represents an intensity of preference in favour of object x as a value of y. Fuzzy sets then represent criteria or flexible constraints. This approach forwarded by Bellman and Zadeh is now fundamental for optimization problems, fuzzy linear programming and decision analysis. Approximate reasoning based on the variables and constraints that can be fuzzy is particularly suitable for using of this concept.

Degree of uncertainty: This interpretation is proposed by Zadeh when he introduced possibility distributions theory. $\mu_A(u)$ is the degree of possibility that a parameter X takes value u. Membership function ranks values in terms of their plausibility. This approach is used in expert systems. When the membership function is defined on this way then the probability that the parameter X takes value u describes as a possibility distribution p_X.

$$p_X = (p_X(u_1)/u_1, p_X(u_2)/u_{2,\ldots,} p_X(u_n)/u_n)$$

Extension of classical relational database model introduced by Codd can be done by including fuzzy values on the attribute domain. These uncertainty information are defined by Zadeh's fuzzy sets and fuzzy logic theory and they allow mathematical framework for representation and handling of imprecise information in fuzzy relational database models.

Fuzzy Relational Database Models

Numerous studies in the field of fuzzy relational database models were introduced in recent years. The literature has reviewed and discussed various issues, such as data representation, different models of the fuzzy relational databases (FRDBMS), the dependence between data, normalization and implementation of FRDBMS and fuzzy query generation. In this paper we present a comprehensive overview of functional dependency analysis which plays an important role in the logical design and database implementation.

Data representation in FRDBMS

Several approaches that include fuzzy information adding in the relational database model are shown in the literature. So, at the first level, fuzzy relational database model is based on the similarity relation. The second group is FRDBM based on fuzzy relation. The most important approach utilizes possibility distribution. The existing approach at this level can be grouped in two classes: attribute value associated with the possibility distribution in the first case, while in another one n-tuple belongs to relation with grade of membership m.

Therefore, we must define a framework for representing imprecise information. Several extensions have been brought to the relational database model to capture the uncertain parts of the real world. This chapter presents four frames for fuzzy representation of data in FRDBMS [9-12]:

- basic framework based on similarity relation,
- basic framework based on possibility distribution,
- basic framework based on fuzzy relation and
- basic framework with extended possibility distribution.

Let R be the relation scheme **R**(Name, Address, Age, Productivity, Salary) and T1, T2, T3 instances at some point of time:

T1: (Mark, Boulevard revolution Str., {21,22,23}, good, high or medium)

T2: (John, New Belgrade, {0.7/22, 1 / 25, 0.8/28}, excellent, {low, medium})

T3: (Peter, Knez Mihail page, 27, satisfactory, high): μ, where $m \in [0,1]$.

In presented model, domain of the attributes can be linguistic terms (low, medium, high, satisfactory...), fuzzy sets (0.7/23, 1/25, 0.8/28), subsets of the given domain (low, medium). Then, we notice that n-tuple may belong to a given relation with some degree of membership.

The basic framework based on similarity relation (Buckles and Petry) provides that each domain of set of attributes in fuzzy relational databases is associated with similarity relation, rather than identity relation and value domain is defined as a subset of the basic set instead a one element as we can see in classical relation databases. Thus, we have following definition:

Definition 1. A fuzzy relation R is a subset of Cartesian product $2^{D1} \times 2^{D2} \times ... \times 2^{Dn}$ where D is finite domain and 2^{Di} is power set of D_i. Any member of the relation is simply called tuple.

A fuzzy relational database is defined as a set of relations where each relation is a set of tuples. According that, fuzzy tuple t_i has the form $t_i = (d_{il}, d_{i2},..., d_{in})$ where $d_{in} \subset D_i$, $d_{in} \neq \emptyset$.

Definition 2. A similarity relation is a mapping s_j: $D_j \times D_j \rightarrow [0,1]$ such that for $x,y,z \in D_j$:

$s_j(x,x) = 1$ (reflexivity)

$s_j(x,y) = s_j(y,x)$ (simmetry)

$s_j(x,z) > \max(\min(s_j(x,y), s_j(y,z))$ ***(max-min transitivity).***

For a given domain D_j, the threshold of similarity is defined as:

$$\text{Threshold (Dj)} = \min_{\forall i}\left\{ \min_{x,y \in d_{ij}}[s(x,y)] \right\}$$

We can present an example of this model with information:

T1:(Mark, Boulevard revolution Str., {21,22,23}, good, high or medium).

The basic framework based on possibility distribution extends the classical theory of relational databases allowing the use of fuzzy values for attributes. The fundamental concept of the fuzzy information is that a variable (attribute) is not definied as a specific value. In this context, we use the term possibility distribution where each attribute value is associated with values from the interval [0,1]. Generally, the possibility distribution is identified with membership function. And we say that element d belongs to a fuzzy set F ("Height of people") with degree of membership 0.9, then the possibility that variable X, defined on the domain F, takes the value d, $\pi_x(d)=0.9$.

Definition 3. A fuzzy relation R is a subset of the domain $\Pi(D_1) \times ... \times \Pi(D_n)$, where:

$\Pi(D_i) = \{\pi_{Ai} \mid \pi_{Ai}$ is possibility distribution of A_i on Di$\}$.

Corresponding tuple is given in the form $t_i = (\pi_{A1}, \pi_{A2,...,} \pi_{An})$.

Further, an extra-element e is introduced in this model which stands for the case when the attribute does not not apply to t_i. The possibility distribution can be viewed as a fuzzy restriction:

$\pi_{Ai:} D \cup e \rightarrow [0,1]$.

An example of this model is presented with information:

T2: (John, New Belgrade, {0.7/22, 1 / 25, 0.8/28}, excellent, {low, medium}).

The basic framework based on fuzzy relation is concept introduced by Baldwin. Fuzzy relation is defined as follows:

Definition 4. Fuzzy relation R on $D_1 \times ... \times D_n$ is determined by the membership function:

$\mu_R : D_1 \times ... \times D_n \rightarrow [0,1]$, where D_i is domain of attribute A_i.

General form of the binary relation R on $D_1 \times D_2$ is represented as:

$R = \{\mu_R(u_1,v_1)/(u_1,v_1),..., \mu_R(u_m,v_m)/(u_m,v_m)\}$ in tuple given as:

$R = \{u_1, v_1, \mu_R(u_1,v_1),...,u_m,v_m, \mu_R(u_m,v_m)\}$

where $u_j \in D_1$, $j = 1,2,...,m$ and $v_k \in D_2$, $k = 1,2,...,n$.

This model specifies that a tuple belongs to a given relation with appropriate grade of membership μ, while the individual attribute values needn't be fuzzy or may be a linguistic variable, but they are treated as atomic or one-variable value.

An example of this model is presented with information:

T3: (Peter, Knez Mihail page, 27, satisfactory, high): μ, where μ∈ [0,1].

The basic framework with extended possibility distribution extends the basic framework based on possibility distribution allowing not only distribution of attribute values, but also and proximity relation associated with a given domain. This extension generalizes the classical relational database model. Note that the similarity relations are only special proximity relations in which closeness relationships are reflexive and symmetric. The properties of reflexivity and symmetry are very appropriate for expressing the degree of closeness or proximity between elements of a scalar domain.

Definition 5. A proximity relation is a mapping s_j: $D_j \times D_j \rightarrow [0,1]$ such that for $x,y \in D_j$:

$s_j(x,x) = 1$ (reflexivity)

$s_j(x,y) = s_j(y,x)$ (simmetry)

In this way, the above-mentioned frameworks become special cases of the basic framework with extended possibility distribution.

An example of this model is presented with information:

T2: (John, New Belgrade, {0.7/22, 1 / 25, 0.8/28}, excellent, {low, medium}).

GEFRED

In the previous years, some authors [1,2,13-16] have dealt with the issue of introducing imprecision and uncertainty information in relational databases. This leads us to the database systems which lie within the scope of artificial intelligence, because they enable to manage information which are very similar to natural language. Codd introduced the relational database organization that is based on relational theory. Zadeh's fuzzy set theory is a generalization of the general theory, while the fuzzy relation concept is generalization of the relational theory.

In this paper we review a general extension of the relational database model called GEFRED. Other models are considered as particular cases of this model. Group of authors [*] introduced General Fuzzy Relational Database Model (GEFRED) that incorporates elements of previous studies into a single model. In this section we introduce the basic elements of a fuzzy extension of relational model.

GEFRED structure model may be shown as follows:

$$R_{FG} \in (D_{G1,}C_1) \text{ x } \ldots \text{ x } (D_{GN,}C_n),$$

where D_{Gi} is a domain of attributes and C_i "attribute compatibility" which takes a value from the interval [0,1]. In this fuzzy relational model attribute compatibility values are not shown, but in each tuple, attribute value is associated with the appropriate value C_i.

Let us consider the following example that describes the extension of classical relational database model to GEFRED.

TABLE 1. GENERALIZED FUZZY RELATIONAL DATABASE MODEL

Name	Address	Age	Productivity	Salary
Mark	Boulevard Revolution	31	Good	High
Alex	Medakovic	Middle	Satisfactory	10.000
Nes	Karaburma	Young	Bad	9.000
Smith	New Belgrade	Old	Excellent	Low
Volter	Cerak	Young	Good	Medium
Greg	Rakovica	About 28	Excellent	13.600
Mathew	Zarkovo	Between 30 i 35	Satisfactory	10.900

Fig. 1. Membership function for the attribute Age [6]

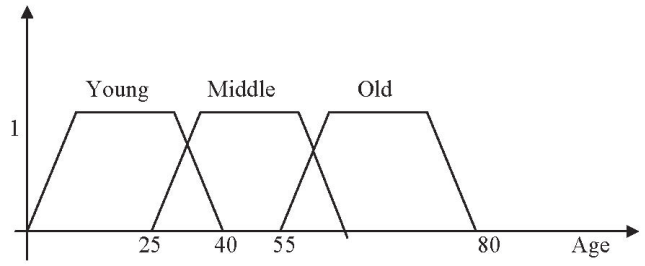
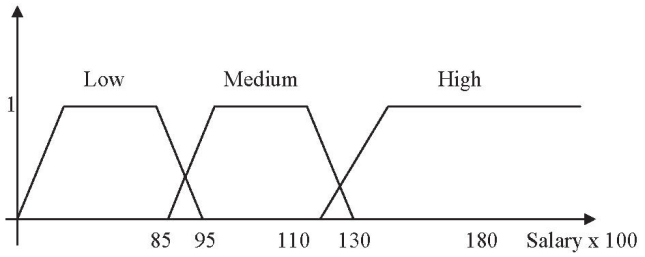


Fig. 2. Membership function for the attribute Salary [6]



The attributes NAME and ADDRESS contain crisp information with the primary key attribute NAME. On the other hand, AGE and SALARY permit fuzzy information and corresponding membership functions for linguistic variables in the relation are shown in the Figs. 1 and 2. The attribute PRODUCTIVITY admists fuzzy information from a discrete domain and we need to define proximity relation over the elements of its domain. This is shown in Table 2.

Table 2. Proximity relation over PRODUCTIVITY

$s(d_i,d_j)$	Bad	Satisfactory	Goog	Excellent
Bad	1	0.85	0.60	0.20
Satisfactory	0.85	1	0.70	0.55
Good	0.60	0.70	1	0.8
Excellent	0.20	0.55	0.80	1

This theoretical model includes all necessary elements for the definition of fuzzy relational database model (FRDBM). Within this database, we can examine the relationship between individual attributes (e.g. salary and productivity). We are able to implement a logical database model knowing functional dependencies. For this reason, in the next section we give a comprehensive overview of functional dependencies analyse between data and attributes in relations.

Data Dependency

Integrity constraints play a key role in the logical database design. Among these limitations, data dependency is the most interesting one because it offers a direct possibility of normalization of relational database model. Therefore, special attention is dedicated to the study of functional dependencies. They bring into relation one set of attribute values with the values of another set of attributes. Based on different models of fuzzy relational databases, different approaches have been proposed for the expression of functional and fuzzy functional dependencies . We differentiate two types of papers in which these topics are studied: the first group includes papers in which the concept of fuzzy functional dependencies is defined, and the other group is consisted of papers in which the concept of functional dependencies for data decomposition and reduction of redundancy and approximation queries in the database is applied.

Functional dependencies

Definition 6. Let R be (A1, A2,..., An) the relational schema to the domains D1, D2,..., Dn with Dom (Ai) = Di and let X and Y be subsets of a set of attributes U = {A1, A2, ..., An} i.e. X, Y ⊆ U and let r be the relation of R, r ⊆ × ... × D1 Dn. We state that the relation r satisfies the functional dependency X → Y if for every two n – tuples t and t'⊆ r, for which t(X) = t' (X) applies, it is implied that it also stands for the t(Y) = t'(Y).

The above mentioned definition indicates that whenever the pairs (x, y) and (x, y ') are elements of R relation [XY] then y = y'. This is precisely the condition that distinguishes the function from the relation. If functional dependency X → Y does not exist, then the relation R [X, Y] can contain multiple elements that have the same attribute value of X, and different attribute value of Y. Secondly, X → Y is a time-invariably ability. A set of n - tuples that describe R (A1,..., An) changes in time, and it is valid for R [X, Y] as well. The definition of functional dependency requires that these changes are such that at any point of time R [X, Y] is not only a function but a relation R [X] → R [Y] as well. The importance of functional dependencies is reflected in the fact that through them we can determine the primary relation

key and what is more important we can define a logical database model.

Fuzzy functional dependencies

If we include the functional dependencies in the fuzzy database model [1.26], the previous definition can not be directly applied to this model because it is based on the concept of equality. Since there is no clear way to verify when the two imprecise values are equal, then the definition of functional dependencies must be extended and generalized respectively. This extended / generalized version of the functional dependency is called the fuzzy functional dependency [1,3,5,15,17,20-22,36]. There are several different definitions of the fuzzy functional dependency, which are obtained as a result of the use of fuzzy logic in the classical functional dependencies and all such definitions of functional dependencies are associated with a given frame in the fuzzy database. Therefore, they are only applicable within a given framework, although there are basic and general features and characteristics that are required to have the fuzzy functional dependency.

In classical relational databases, functional dependencies determine when the value of the n-tuples of one set of X attribute uniquely determines its values in another set of Y attribute or strictly speaking: Generally, when the attribute values do not take on only the domain atomic elements, but also possibility distribution, then the $X \rightarrow Y$ degree is not necessarily required to be 1, but may be in the unit interval [0,1]. Therefore, the following questions are naturally imposed. How to determine the t (X) = t '(X) and t (Y) =t' (Y) when t (X), t '(X), t (Y) and t' (Y) are all imprecise values of possibility distribution. Secondly, how to determine the level of propositions; if t (X) = t '(X) then t (y) = t' (Y) where t (X) = t '(X) and t (Y) = t' (Y) are partially true with the degree of membership from the interval [0,1]. Finally, how to assess the degree of $X \rightarrow Y$ if different pairs of n - tuples give different true values for the proposition if ... then. Hence what arises is that these issues are associated with problems of fuzzy proximity data, fuzzy logic implications and fuzzy (and) operator. Now, let us define the fuzzy functional dependencies in the following way.

Definition7. Let R be (A1, A2,..., An) the relational schema to the domains D1, D2,..., Dn with Dom (Ai) = Di and let X and Y be the subsets of a set of attributes U = {A1, A2, ..., An} i.e. $X, Y \subseteq U$ and let r be the relation of R, $r \subseteq \Pi (D1) \times ... \times \Pi (Dn)$, where $\Pi (Di) = \{\pi \mid \pi$ is a possible distribution of Ai at Di, i = 1,2,..., n.

We say that X fuzzy functionally determines Y with the degree of θ designated as $X \overset{\theta}{\rightarrow} Y$ if and only if for $\forall r \in R$:

$$\min I (t (X) = c\, t'(X), t (Y) = c\, t' (Y)) \geq \theta, t, t' \in R.$$

where $\in\theta [0,1] = C [0,1] \times [0,1] \rightarrow [0,1]$ is a measure of proximity (closeness)

and I: $[0,1] \times [0,1] \rightarrow [0,1]$ is a fuzzy implication operator.

The rules of executing functional dependencies

In classical relational model, it is often necessary, based on a given set of dependent data, to determine some other dependencies on the same database which are the result of already given set of dependencies. Therefore, in the classical relational databases, there are three rules of executing known as Armstrong's axioms, and which are used to derive new functional dependencies from the given functional dependencies. Now, we provide these rules of executing [9]. Let X, Y, Z and W be an arbitrary set of attributes.

P_1: If $Y \subseteq X$ then the functinal dependency exist $X \rightarrow Y$.

P_2: If functional dependency $X \rightarrow Y$ applies then the functional dependency $XZ \rightarrow YZ$ also applies.

P_3: If functional dependencies $X \rightarrow Y$ and $Y \rightarrow Z$ apply then the functional dependency $X \rightarrow Z$ *also applies.*

Fuzzy functional dependencies express the relation between the set of attributes. If the relational schema is given to a set of attributes U and the fuzzy

functional dependency $X \xrightarrow{\theta} Y$, which is satisfied in R, then $X \xrightarrow{\theta} Y$ signifies that it is satisfied in all the relations of R. Moreover, for a given set of fuzzy functional dependencies F, which is satisfied in F, does not only guarantee that these fuzzy functional dependencies of F are met in all relations of R, but it also guarantees that each fuzzy functional dependency, logically implied with F, is satisfied in all relations of R. For example, if we know that $X \xrightarrow{\theta} Y$ in R applies, then we can expect that in R also applies $X \xrightarrow{\alpha} Y$ for a $\leq \theta$ from [0,1]. Also, if $X \xrightarrow{\theta} Y$ applies then we can expect that XZ functionally determines YZ to a degree that is at least equal to θ, for all relations of R. Furthermore, if $X \xrightarrow{\alpha} Y$ and $Y \xrightarrow{\beta} Z$ apply in R then we can expect that X functionally determines Z with some degree of λ from [0.1]. Intuitively, we can assume that $\lambda = \min(a, \beta)$ for the following reasons:

a. If X functionally determines Y with the degree of θ and Y functionally determines Z with the degree of θ, then we can expect that X functionally determines Z with the same degree.

b. If X functionally determines Y with the degree of $\alpha \geq \theta$ and Y functionally determines Z with the degree of $\beta \geq \theta$, then we expect that X functionally determines Z with the degree of θ. Thus, the above given three Armstrong's axioms in the classical relational theory are expanded with three rules of executing fuzzy functional dependencies:

   P'1: If $Y \subseteq X$ then there is fuzzy functional dependency $X \xrightarrow{\theta} Y$ for $\forall\theta$.

   P'2: If fuzzy functional dependency $X \xrightarrow{\theta} Y$ applies, then the fuzzy functional dependency $XZ \xrightarrow{\theta} YZ$ also applies.

   P'3: If fuzzy functional dependencies $X \xrightarrow{\alpha} Y$ and $Y \xrightarrow{\beta} Z$ apply, then the fuzzy functional dependency $X \xrightarrow{\lambda} Z$ *with* $\lambda = min(\alpha,\beta)$, where $\lambda, \alpha, \beta, \in\theta [0,1]$.

Analysis models of functional dependencies

In the field of functional and fuzzy functional dependencies we point out the following papers [2,11,13,14,21, 23,24,30,32-35].

Fuzzy functional dependencies in Raju and Majumdar's model

Raju and Majumdar's [30] fuzzy relational model allows components of n- tuples to take both atomic and non-atomic values. Depending on the complexity of the domain, they divide fuzzy relation into two categories. In the first type of the fuzzy relation, domain can only be a fuzzy set or classical set. The second type of the fuzzy relation ensures that each attribute domain can be a classical set, fuzzy set or a set of fuzzy subsets (or possibility distribution). Each fuzzy relation in this model is represented by a table that has an additional column which determines the membership value of a given n – tuples to an appropriate relation.

Raju and Majumdar define the fuzzy functional dependencies in the following way:

Definition 8. Fuzzy functional dependency $X \rightarrow Y$, $X, Y \subseteq R$, applies in fuzzy relation r on R, if for all n-tuples t_1 and t_2 from r ($m_r(t_i) > 0, i = 1,2$), applies:

$$\mu_{EQ}(t_1[X], t_2[X]) \leq \mu_{EQ}(t_1[Y], t_2[Y])$$

where Equal (EQ) is a fuzzy relation of proximity (closenesss) on a universal set U and is defined as a fuzzy subset on $U \times U$ and where μ_{EQ} is a membership function that satisfies the following conditions: For every $a,b \in U$, $\mu_{EQ}(a,a) = 1$ (reflexive) and $\mu_{EQ}(a,b) = \mu_{EQ}(b,a)$ (symmetric).

Fuzzy functional dependencies in Saxena and Tyagi's model

In the fuzzy relational model of Saxena and Tyagi [32] fuzzy attribute values are allowed the possibility where the attribute is not applicable to a given object. To be able to work with vague data values, they define the fuzzy relation as follows. Let 2^{A_i}, i =

1,2,...,n be a set of fuzy subsets on the domain dom $A_i \cup \{e\}$, where e is the extra element which allows the opportunity for a fuzzy attribute value not to be applied to a given object, such that for each set of $a \in 2^{Ai}$, a membership function from a satisfies the condition $\mu_a(e) = 0$ ili $\mu_a(e) = 1$ of the set with $\mu_a(u) = 0$ for each u. At that point, the fuzzy relation r on $R = (A_1...,A_n)$ is defined as a fuzzy subset of the Descartes' product $2^{A1} \times 2^{A2} \times ... \times 2^{An} = 2^R$, characterized by a membership function:

$$\mu_r: 2^R \rightarrow [0,1].$$

Every n - t = tuple $t = (a_1,...,a_n)$ where $a \in 2^{Ai}$, i = 1,2,...,n in r can be seen as possibility distribution for $D = (\text{dom } A_1 \cup \{e\}) \times \ldots \times (\text{dom } A_n \cup \{e\})$ as:

$$Poss(t[A_1] = u_1, ..., t[A_n] = u_n) = \min \{\mu_r(t), \mu_{a1}(u_1),..., \mu_{an}(u_n)\},$$

where $u_i \in A_i, \cup \{e\}$, i = 1...n.

In this fuzzy relational model, the similarity between the elements of a given domain is defined as follows.

$$\mu_{EQ}(a_1,a_2) = \min_{u \in dom A_i} \psi(\mu_{a1}(u), \mu_{a2}(u))$$

Here the fuzzy relation is EQ na 2^A-{Ø}, where, with Ø it is marked that for the attribute A, a fuzzy set $a \in 2^A$ is such that $\mu_a(e) = 1$ *and* $\mu_a(u) = 0$ is *for* $u \neq e$, and is defined as a fuzzy subset $(2^A\text{-}\{Ø\}) \times (2^A\text{-}\{Ø\})$ so that its membership function is defined as $\mu_{EQ}(2^A\text{-}\{Ø\}) \in (2^A\text{-}\{Ø\}) \rightarrow [0,1]$ and which meets the $\mu_{EQ}(a,a) = 1$ (reflexive), $a \in 2^A\text{-}\{Ø\}$ and $\mu_{EQ}(a_1,a_2) = \mu_{EQ}(a_2,a_1)$ (symmetric), $a_1, a_2 \in 2^A\text{-}\{Ø\}$. ψ is the fuzzy relation of similarity. Based on the above considerations, in this model, they introduce fuzzy functional dependencies.

Definition 9. Fuzzy functional dependency $X \rightarrow Y$, $X, Y \in R$, applies in fuzzy relation r on R, if for any n-tuples t_1 and t_2 from r $\mu_r(t_i) > 0$, i = *1,2* applies $\mu_{EQ}(t_1[X],t_2[X]) > 0$, $t_1[Y] = t_2[Y] = Ø$ or a non-empty set exists $Y' \subseteq Y$ such that $t_1[A] \neq Ø \neq t_2[A]$ for every $A \in Y'$, $t_1[Y - Y'] = t_2[Y - Y'] = Ø$ *and* $\mu_{EQ}(t_1[X], t_2[X]) \leq \mu_{EQ}(t_1[Y'], t_2[Y'])$.

Fuzzy functional dependencies in Wei-Yi Liu model

This paper first defines the concept of semantic distance between two fuzzy values of attributes, while the fuzzy functional dependencies are represented by the fuzzy semantic distance. Based on the semantic proximity, definition of fuzzy functional dependencies is given [21,23,24]. The degree of closeness between the two fuzzy values is described by means of semantic proximity. Semantic proximity is based on the concept of the interval and we mark it with the $SD(f_1,f_2)$, where $0 \leq SD(f1, f2) \leq 1$. The following characteristics should be met:

If $f_1=[a_1,b_1]$, $f_2=[a_2,b_2]$, $g_1=[c_1,d_1]$, $g_2=[c_2,d_2]$. Then applies:

1. $SD(f_1,f_2)=1$ if and only if $a_1 = a_2 = b_1 = b_2$,

2. $SD(f_1,f_2)=0$ if and only if $f_1 \cap f_2 = Ø$,

3. If $a_1 = a_2$, $b_1 = b_2$, $c_1 = c_2$, $d_1 = d_2$ and $|d_1 - c_1| > |b_1 - a_1|$ then $SD(f_1,f_2) \geq SD(g_1,g_2)$,

4. If $|a_2 - b_2| = |a_1 - b_1|$ and $f_1 \cap g_1 \geq f_2 \cap g_1$ then $SD(f_1,g_1) \geq SD(f_2,g_1)$.

Semantic distance (proximity) is calculated by the following formula:

$$SD(f_1,f_2) = ||f_1 \cap f_2|| / ||f_1 \cup f_2|| - ||f_1 \cap f_2|| / \alpha$$

where $||h||$ is modular with an appropriate interval:

$$\| h | = \begin{cases} 0 & h \neq 0 \\ \delta & h = [a,a] \\ |b - a| & h = [a,b] \\ \alpha & h = \infty \end{cases}$$

α is such a coefficient that $\alpha \geq ||f_1 \cup f_2||$, and δ is a small number. E.g. $\delta = \alpha/10.000$ is usually taken in concrete examples.

If $t_1 = (x_{11}, x_{12}, ..., x_{1n})$ and $t_2 = (x_{21}, x_{22}, ..., x_{2n})$ are two n-tuples in relation, then the semantic proximity between them is marked as $SD(t_1,t_2)$ and calculated

as $SD(t_1,t_2)=\min\{SD(x_{1i},x_{2i})\}$.

The semantic distance $SD(f_1,f_2)$ of the two fuzzy values f1 (X) and f2 (X) is defined by using a certain standard $|f_1(X) - f_2(X)|$. For example, $SD(f_1,f_2) = max\ |f_1(X) - f_2(X)|,\ x \in dom(A_i)$. The complement of $SD(f_1,f_2)$ in the $SS(f_1, f_2)$ mark is defined as $SS(f_1, f_2) = 1 - SD(f_1, f_2)$. The semantic distance of the fuzzy values described in another way can be defined similarly.

Definition 10. Let r be the fuzzy relation on the relational schema $R(A_1,...,A_n)$ and let U be a universal set of attributes $A_1,...,A_n$ and let X and Y be subsets of U. We say that fuzzy relation r meets fuzzy functional dependency $X \rightarrow Y$, $X,Y \subseteq R$ if for each pair of n-tuples t_1 and t_2 in r applies:

$$SS(t_1[X], t_2[X]) \leq SS(t_1[Y], t_2[Y])$$

Fuzzy functional dependencies in Dubois-Prade model

Dubois and Prade [13,14] introduce fuzzy functional dependencies as follows: if the attribute A values for the n-tuples t and t' are equal then the attribute B values for t and t' should not be far away from each other. They model this idea by expressing the fuzzy relations of closeness P (which is reflexive $\forall d,\ \mu_p(d,d) = 1$ and symmetric $\forall d,d'\ \mu_p(d,d') = \mu_p(d',d)$), and which is defined in the field of attribute B meaning.

If $t(A) = t'(A)$ then $\mu_B\ (t(B),t'(B)) > \theta$

where θ represents inflicted threshold.

Considering that t(B) i t'(B) is determined in the term of possibility distribution fuction, we have:

If $t(A) = t'(A)$ then $\Pi\ (t(B) \approx_p t'(B)) > \theta$

This possibility is given with:

$$\Pi\ (t(B) \approx_p t'(B)) = \sup_{v,w \in D_B \times D_B} \min\ (\mu_p(v,w),\ \pi_{t(B)}(v),\ \pi_{t'(B)}(v))$$

where π_B is a function of possibility distribution which limits possible meaning of the attribute B for the n-tuple t, and μ_p is a membership function of the closeness P fuzzy relation.

Fuzzy functional dependencies in Shenoi-Melton model

Their strategy for the expression of imprecise information is based on the idea of collection of congenial elements of the final domain into blocks of elements that do not differ from a certain level of accuracy in that domain. This idea is expressed in relation to the classes of equivalence at the domain of partition. The partition of the domain Dk is the set of non-empty disjoint subsets or equivalence classes of Dk with the property that each element from Dk is exactly in one equivalence class. [33.34] Partition of the domain into classes of equivalence is the key to the preservation of some important features in the classical model. On this basis they define redundancy as follows:

Definition 11. Let t and t' be two fuzzy n-tuples. Components t_k and t'_k are α_k – redundantly marked as $t_k \approx_{\alpha k} t'_k$, when t_k and t'_k are subsets of the same equivalence class for α_k – partititions of the temporal domain D_k.

They define fuzzy relation as follows:

Definition 12. Let R be the relational schema with attributes $(A_1...,A_n)$ and adjoint partitions with levels $(\alpha_1,..., \alpha_n)$. Let r be $(R) \subseteq 2^{D1} \times 2^{D2} \times ... \times 2^{Dn}$. Then r is the fuzzy relation in relation to the fuzzy relational schema R and temporal domain $D'_1,...,D'_m$, if r is a set of non-redundant fuzzy n-tuples with respect $\alpha_1,...,$ α_n – partition level on $D'_1,...,D'_m$, respectively.

Definition 13. Let R be the fuzzy relational schema with attributes U and level of partition of α_U. Let X and Y be subsets of attributes in U with associated levels $\alpha_X = (\alpha_p,..., \alpha_q)$ and $\alpha_Y = (\alpha_r,..., \alpha_s)$ in α_U. Let r be the fuzzy relation with temporal domains $D'_p,...,D'_q$, and $D'_r...,D'_s$, for subsets of attributes X and Y respectively. Relation r meets the fuzzy functional dependency from X in Y with levels (α_X,α_Y) on given partitions, when for any n-tuples t, $t' \in r(R)$ applies:

$t \approx_{\alpha x} t'$ implies $t \approx_{\alpha y} t'$

Finally, they argue that such defined fuzzy functional dependencies satisfy Armstrong's axioms.

Fuzzy functional dependencies in Sozati and Yazici model

One of the most fundamental definitions of the fuzzy functional dependencies is given in paper [35]. The analysis of the fuzzy functional dependencies is based on the following considerations. If t [X] is similar to t '[X], then t [Y] is also similar to the t' [Y]. In fact the similarity between the Y values is greater than or equal to the similarity between the X values. Such dependency is marked as $X \underset{F}{\rightarrow} Y$.

An example of such dependency is a dependency "officers with similar experiences should have similar salary".

In this case, attribute values of Experience and Salary can be imprecise, while the definition of dependency is strictly defined. However, this definition of functional dependency is not fully determined, in the sense that the dependency itself can be imprecise. An example of such functional dependency is a "level of person's intelligence *more or less* determines his/her success", where *more or less* in the sentence determines imprecise dependency. If we know that the person is intelligent, then we can conclude that he/she will be successful. However, this level of success is not clearly nor precisely determined by intelligence. A person can be very successful, less successful, and so on. Thus, this dependency does not determine the precise level of success, but at least it ensures a minimum level of success. Suppose there are two people with an identical intelligence and suppose that the first person is very successful. From this, one can not conclude that the other person is very successful, but we can say that the level of success of the other person will be more or less similar to the level of success of the first person.

One way to define this kind of dependency is to accept the linguistic intensity of dependency as a threshold. For example, dependency "officers with similar experiences should have similar salary" has a linguistic strength of 1, while dependency "The level of intelligence more or less determines the success" has a linguistic strength of 0.7. Thus, the threshold value determines the dependency intensity (strength) and it is written in the form of $X \underset{F}{\overset{\theta}{\rightarrow}} Y$, where θ is the dependency intensity (strength). The concept of similarity is very important in fuzzy relational databases because it allows us to extend the concept of identity into a clear model for handling imprecise and uncertain information. In "Crisp" models of databases, two n-tuples are identical on the observed attribute if and only if the values of that attribute are identical. In the fuzzy models of databases, the similarity of attribute values is observed in the sense to which extent these values are adjusted on the observed attribute. In this fuzzy model, the similarity between the attribute values is defined as the conformance of the two n-tuples on the attribute. This is very important aspect proposed by Bosc and others which is used for the comparison of the values of imprecise, fuzzy attributes by using the concept of conformance. The conformance relation is symmetric, reflexive and transitive.

Definition 14. The attribute conformance A_k is defined on the domain D_k for any n-tuples t_i and t_j, presented in relation r and marked as $C(A_k[t_i,t_j])$, is given as:

$$C(A_k[ti,tj]) = \min\left\{\min_{x\in d_i}\left\{\max_{y\in d_j}\{s(x,y)\}\right\}, \min_{x\in d_j}\left\{\max_{y\in d_i}\{s(x,y)\}\right\}\right\},$$

where d_i is the attribute value of A_k for n-tuple t_i, d_j is the attribute value of A_k for n-tuple t_j, s(x,y) is the similarity relation for values x and y, and s is a mapping of each pair of elements from the domain D_k in the interval [0,1],

If $C(A_k[t_i,t_j]) > \theta$, for relation r, for n-tuples t_i and t_j we state that they are agreeable on the attribute A with the dependency intensity of θ. This definition is extended to the description of closeness for two n-tuples on the set of attributes.

Definition 15. The conformance on the set of the attribute X for any two n-tuples t_i and t_j, given in relation r, marked as $C(X[t_i,t_j])$, is given as:

$$C(X[t_i,t_j]) = \min_{A_k \in X} \{C(A_k[t_i,t_j])\}$$

Definition 16. Let r be the fuzzy relation on the relational schema $R(A_1,...,A_n)$ and let U be a universal set of attributes $A_1,...,A_n$ and let X and Y be the subsets of U. We state that the fuzzy relation r meets the fuzzy functional dependency $X \xrightarrow[F]{\theta} Y$, if for each pair of n-tuples t_i and t_j in r applies:

$$C(Y[t_i,t_j]) \geq \min(\theta, C(X[t_i,t_j]))$$

where θ is a realistic number from [0,1] and describes the linguistic strength of dependency.

Fuzzy functional dependencies in Cubero – Medina model

The theory of normalization which has been introduced by Codd is a systematic approach to a proper designing of databases. The main idea is that if we are faced with relations in databases which satisfy functional dependency (excluding the primary key), then there is a possibility of redundancy and updating of the existing base. In order to avoid redundancy, we can decompose the original relation, i.e. create decomposition, without having lost any information. Normally, in real databases, it is not usual that strict dependencies in relations are given. Nevertheless, we can find functional and fuzzy functional dependencies such as „The weight of a person more or less depends on his/her height and age". In these situations, the process of decomposition is proposed [11] and extraction of information respecting given dependency and compression of original data in relational database. The idea represents the use of fuzzy set theory and tolerance towards some uncertainties in the base, which allows us to include more n-tuples into one. Let us consider for example the relation which appears in the following relation. A special association (mapping) operator is used for the recovering of original data from the R relation.

TABLE 3. ORIGINAL RELATION R [11]

X	Height	Weight
X1	180	86
X2	170	74
X3	170	73

TABLE 4. THE INTRODUCTION OF FUZZY VALUES IN RELATIONAL STRUCTURE[11]

X	Height	Weight
X1	High	ca 85 kg
X2	170	74 kg
X3	170	ca 73 kg

TABLE 5. RELATION R_1 [11]

X	Height
X1	180
X2	170
X3	170

TABLE 6. RELATION R_2 [11]

Height	Weight
High	ca 85 kg
170	74 or ca73 kg

The decomposition of the relation R is given by relations (projections) r1 and r2, as shown in the previous tables. As can be seen in this example, we have reduced redundancy because the second and third tuple are merged into one in relation r2. In such situations, it is of primary importance to quantify how much of the imprecision we can tolerate, in order to guarantee that the fuzzy values such as "Ca 85 kg" are close enough to the original data of 86 kg. In order to do this, we must use the measure of similarity between data elements. For the new data in the base, we can test the fuzzy dependency by observing n-tuples in relation r2. In this way, the amount of data stored in databases r1 and r2 is smaller than in relation R. The original data that appears in R can be obtained by merging of relations r1 and r2. It should be noted that linguistic variables, such as "Ca 85 kg" are defined and given by the experts - experienced database managers. Therefore, we should be able to discover some knowledge in the form of the fuzzy rules that will allow us, that after decomposition of the original relation, we reduce and remove redundancy. This certainly allows us to get a better understanding of the real world, because fuzzy dependencies are isolated in a special relation.

The definition of fuzzy functional dependency in the Cubero – Medina model:

Definition 17. If $R_i(X_i(t_1),X_i(t_2)) \geq \alpha_i \ \forall i$ then $R_j(Y_j(t_1),Y_j(t_2)) \geq \beta_j \ \forall j$ must apply. As a special case we have $\forall\ t_1, t_2 \in r$ where apllies that $X_i(t_1)=X_i(t_2) \ \forall i$, then $R_j(Y_j(t_1),Y_j(t_2)) \geq \beta_j \ \forall j$, for the case of the existence of fuzzy functional dependencies.

So, Cubero, Medina and others introduce other fuzzy functional dependencies in the relational database model. This approach allows us to discover connections between the attributes that are not detected by the classical approximation, and execute decomposition respecting established fuzzy dependencies. In this way, we reduce the redundancy in the databases, save computer resources, while at the same time there is no loss of information. The disadvantage of this approach is the fact that for efficient design and establishment of the fuzzy dependencies the help of experts and experienced database managers is necessary.The solution for this kind of problem could be data mining systems, i.e. the design of efficient algorithms for detecting the fuzzy functional dependencies without involving subjective human factor.

Fuzzy approximate dependencies

Within the analysis of data in relational databases, a very interesting question is detecting possible relations between attribute values, and at a higher level the relation between the attributes themselves, respectively, i.e. the analysis of functional and multivalued dependencies. In the case of the presence of uncertainty and vagueness of data, specific methods of data mining techniques are used in knowledge discovery. Berzal, Blanco and others [2] propose an algorithm for computing approximate fuzzy dependencies and different types of relations between attributes in the fuzzy relational database models.

In real databases, we are faced with two various types of relations. On the one hand, there are relations that are implicit, in which the relations between the attribute values are hidden and which are not clear enough at first. This type of dependency is obtained through the analysis of the database itself. On the other hand, we are often faced with the explicit relations between attributes that are easily detected (e.g. City and Zip Code). These two types of relations between attributes in the relational database structures represent integrity constraints that are imposed in the process of database design. In these cases we argue that there is a functional dependency or approximate dependency between attributes.

Search for functional dependencies in relational databases is a subject of interest in the field of data mining, as this form of business intelligence strictly deals with the structure of data. However, it is very difficult to perfectly detect functional dependencies in databases as a single exception in the rules affects the loss of dependency. If a number of these exceptions is not large, "fuzzy functional dependencies with exceptions" can indicate interesting regularities contained in the data. Moreover, the level of dependency which exists between the data is determined and presented. The idea is to measure not only the accuracy of dependency, but also support (the proportion of n-tuples in which the observed dependency occurs). Therefore, for the dependency assessment, Confidence is used - the conditional probability p (Y / X), written as Conf $(X \rightarrow Y)$ and support (Support) - The probability $p(X \cup Y)$, written as S $(X \rightarrow Y)$.

The problem with Confidence is the fact that it does not take into account the negative dependencies, therefore, high percentages of confidence can be obtained, which in these cases can be misleading. Therefore, in papers, the use of the safety factor CF is proposed:

$$CF(X \rightarrow Y) = \begin{cases} \dfrac{Conf(X \rightarrow Y) - S(Y)}{1 - S(Y)}, & Conf(X \rightarrow Y) > S(Y) \\ \dfrac{Conf(X \rightarrow Y) - S(Y)}{S(Y)}, & Conf(X \rightarrow Y) < S(Y) \\ 0 & Inace \end{cases}$$

Safety factor takes values from the interval [-1,1] and shows us to which extent is our conviction of the dependency existence true. CF= 1 in situations where X =True then Y = True, and CF =- 1 otherwise. Two extreme cases are when S (Y) = 0 and S (Y) = 1. In both cases the result is trivial, therefore it is logical that we then take the value of CF = 0.

Definition 18. If CF $(X \rightarrow Y) = 1$ (which implies that Conf $(X \rightarrow Y) = 1$) then $X \rightarrow Y$ is a functional dependency.

Conclusion

Inclusion of fuzzy information in different models of databases is an important research topic because fuzzy data is intensively present in a number of applications that we face and work with. The very essence of this kind of data models is the fact that there are many active research areas that directly include or use these knowledge bases, such as: geographic information systems (GIS) and spatial data representation, data Mininig systems, fuzzy information search, the statistical database models.

In this paper, a survey of different approaches in the analysis of functional dependencies which play a key role in designing and logical database designing has been conducted. Various fuzzy models based on the analysis of data dependency have been proposed in the last two decades and there is a significant number of papers and a large number of authors who deal with this issue. As we have seen, there are several frameworks for defining the functional and fuzzy functional dependencies, and which are more or less based on the similarity relation between the elements of a given domain. For all of them there are appropriate rules of executing, which demonstrate when from a given set of fuzzy dependency (functional or fuzzy functional) other dependencies are logically derived. In order for the new fuzzy dependencies to be derived from a given set of fuzzy dependency, there must be appropriate axioms made for them, which are based on Armstrong's axioms for the classical dependency. All these dependencies and rules of deduction must satisfy the adequacy requirement (sound) and completeness (complete).

However, it is noticed that the test dependency procedure and derivation of the logical consequences from a given set of attributes is very complex. Practically, there is no efficient algorithm that would enable us to easily identify the dependencies between the observed set of attributes and application of the normalization theory. Therefore, the subject of future studies is defining the framework and application of different mathematical tools that will enable simpler identification and discovery of knowledge necessary for the elimination of redundancy and different types of anomalies.

References:

[1] Belohlavek, R., (2008) *Codd's Relational Model from the Point of View of Fuzzy Logic.*

[2] Berzal, F. I. Blanco, D. Sánchez, J.M. Serrano, M.A. Vila, (2005), "A definition for fuzzy approximate dependencies", *Fuzzy Sets and Systems*, Vol. 149, pp. 105-127.

[3] Bhatt, Rajen B., M. Gopal, (2006), "On the extension of functional dependency degree from crisp to fuzzy partitions", *Pattern Recognition Letters 27*, pp. 487–491.

[4] Bosc, P., Donald Kraft, Fred Petry, (2005), "Fuzzy sets in database and information systems:status and opportunities", *Fuzzy Sets and Systems*, Vol.156, pp. 418-426.

[5] Bosc, P., Olivier Pivert, (2003), "On the impact of regular functional dependencies when moving to a possibilistic database framework", *Fuzzy Sets and Systems*, Vol. 140, pp. 207-227.

[6] Bosc, P., M.Galibourg, (1989), "Indexing principles for a fuzzy database" *Information Systems*, Vol. 14, pp. 493-499

[7] Buckles, B. P. Frederick E. Petry, (1984), "Extending the fuzzy database with fuzzy numbers", *Information Sciences*, Vol. 2, pp. 145-15.

[8] Buckles, B.P. Petry, F. E., (1982), "A fuzzy representation of data for relational databases", *Fuzzy Sets and Systems*, Vol. 7, pp. 213-226.

[9] Chen, G., (1998) Fuzzy logic in data modeling: semantics, constraints and database design, Kluwer Academic Publisher.

[10] Chiang, Ding-An. Louis R. Chow, Nan-Chen Hsien, (1997), "Fuzzy information in extended fuzzy relational databases", Fuzzy Sets and Systems, Vol. 92, pp. 1-20.

[11] Cubero, J. C. J. M. Medina, O. Pons, M. A. Vila, (1999), "Data summarization in relational databases through fuzzy dependencies", Information Sciences, Vol. 121, pp. 233-270.

[12] Dubais, D., H.Prade, (1980), Fuzzy sets and systems: theory and application, Academic Press, New York.

[13] Dubois, D., Henri Prade, (1997), "The three semantics of fuzzy sets", Fuzzy Sets and Systems, Vol. 90, pp. 141-150.

[14] Dubois, D., Henri Prade, (2003), "Fuzzy set and possibility theory-based methods in artificial intelligence", Artificial Intelligence , Vol. 148, pp. 1–9.

[15] Haris, J., (2006), Fuzzy logic application in engineering science, Springer.

[16] Hussai, T., Mian M. Awais, Shafay Shamail, (2006), "Applying fuzzy logic to measure completeness of a conceptual model", Fuzzy Sets and Systems.

[17] Jyothi, S., M. Syam Babu, (1997), "Multivalued dependencies in fuzzy relational databases and lossless join decomposition", Fuzzy Sets and Systems, Vol. 88, pp. 315-332.

[18] Kacprzyk, J., Bill P. Buckles, Frederick E. Petry, (1990), "Fuzzy information and database systems", Fuzzy Sets and Systems, Vol. 38, pp. 133-135.

[19] Kaufman, A. (1985), An introduction to the theory of fuzzy subsets, Academic Press, New York.

[20] Kraft, D., H. Frederick E. Petry, (1997), "Fuzzy information systems: managing uncertainty in databases and information retrieval systems", Fuzzy Sets and Systems, Vol. 90, pp. 183-191.

[21] Liu, W., (1997), "Fuzzy data dependencies and implication of fuzzy data dependencies", Fuzzy Sets and Systems, Vol. 92, pp. 341-348.

[22] Liu, W., (1993), "Extending the relational model to deal with fuzzy values", Fuzzy Sets and Systems, Vol. 60, pp. 207-21.

[23] Liu, W., (1994), "Constraints on fuzzy values and fuzzy functional dependencies", Information Sciences, Vol. 78, pp. 303-309.

[24] Liu, W., (1997), "A relational data model with fuzzy inheritance dependencies", Fuzzy Sets and Systems, Vol. 89, pp. 205-213.

[25] Ma, Z.M., Li Yan, (2008), "A literature overview of fuzzy database models", Information Science and Engineering, pp. 172-180.

[26] Ma, Z., (2006), Fuzzy database modeling of imprecise and uncertain engineering information, Springer.

[27] Medina, J. M.l, Olga Pons, Maria Amparo Vila, (1994), "GEFRED: A generalized model of Fuzzy Relational Databases", Information Sciences, Vol. 76, pp. 87-109.

[28] Medina, J. M., M. A. Vila, J. C. Cubero, O. Pons, (1995), "Towards the implementation of a generalized fuzzy relational database model", Fuzzy Sets and Systems, Vol.75, pp. 273-289.

[29] Radecki, T., (1983), "A theoretical background for applying fuzzy set theory in information retrieval", Fuzzy Sets and Systems, Vol. 10, pp. 169-183.

[30] Raju, K. V. S. V. N. Arun K. Majumdar, (1987), "The study of joins in fuzzy relational databases", Fuzzy Sets and Systems, Vol. 21, pp. 9-34.

[31] Rundensteiner, E. A. Lois W. Hawkes, Wyllis Bandler, (1989), "On nearness measures in fuzzy relational data models", International Journal of Approximate Reasoning, Vol. 3, pp. 267-298.

[32] Saxena, P. C., B. K. Tyagi, (1995), "Fuzzy functional dependencies and independencies in extended fuzzy relational database models", Fuzzy Sets and Systems, Vol. 69, pp. 5-89.

[33] Shenoi, S., Austin Melton, L.T. Fan, (1992), "Functional dependencies and normal forms in the fuzzy relational database model", Information Sciences, Vol. 60, pp. 1-28

[34] Shenoi, S., Austin Melton, L. T. Fan, (1990), "An equivalence classes model of fuzzy relational databases", Fuzzy Sets and Systems, Vol. 38, pp. 153-170.

[35] Sözat, M., Adnan Yazici, (2001), "A complete axiomatization for fuzzy functional and multivalued dependencies in fuzzy database relations", Fuzzy Sets and Systems, Vol. 117, pp. 161-181.

[36] Vandenberghe, R. A. Van Schooten, R. De Caluwe, E. E. Kerre, (1989), "Some practical aspects of fuzzy database techniques: An example", Information Systems, Vol. 14, pp. 465-472.

Submitted: February 07, 2011
Accepted: December 19, 2011

Mutation Testing: Object-Oriented Mutation and Testing Tools

Z. Ivanković, B. Markoski, D. Radosav

University of Novi Sad, Technical Faculty "Mihajlo Pupin", Zrenjanin, Serbia

zdravko@tfzr.uns.ac.rs

Contribution to the State of the Art

UDC **004.7:006**

Abstract: Software testing represents activity in detecting software failures. Mutation testing represents a way to test a test. The basic idea of mutation testing is to seed lots of artificial defects into the program, test all defects individually, focus on those mutations that are not detected, and, finally, improve the test suite until it finds all mutations. Mutants can be created by mutating the grammar and then generating strings, or by mutating values during a production. Object-oriented (OO) programming features changed the requirements for mutation testing. Non object-oriented mutation systems make mutations of expressions, variables and statements, but do not mutate type and component declarations. OO programs are composed of user-defined data types (classes) and references to the user-defined types. It is very likely that user-defined components contain many defects such as mutual dependency between members/classes, inconsistencies or conflicts between the components developed by different programmers. Class Mutation is a mutation technique for OO programs which particularly targets plausible faults that are likely to occur due to features in OO programming. Mutation testing requires automated testing tools, which is not a trivial tool to make. Automated mutation tools must be able to parse the program and know its language. When the program is run, mutant can be killed by one of two possible scenarios: if a mutant crashes, or if the mutant goes into an infinite loop.

Key words: Mutation testing, Object-oriented mutation, schema-based mutation, reflection

Introduction

Software testing represents activity in detecting software failures. The scope of software testing often includes examination of code as well as execution of that code in various environments and conditions as well as examining the aspects of code: does it do what it is supposed to do and do what it needs to do. Software testing is faced with several problems. Bugs are not distributed uniformly across a program: "20% of modules contain 80% of the defects". Second problem represents risk, which is unevenly distributed. In every project there are some modules in which defects have serious consequences because they are frequently used, or because entire functionality depends on them. Tester would want his test suite to be focused on the defect-prone modules, and to make his testing efforts based on the risk, rather than achieving a specific coverage.

Many systems are tested according to principle: "if it does not crash, it is probably fine". In this way, test does not check the result, and a tester cannot determine how well a test does its job. This is an instance of Plato's old problem: "Who watches the watchmen". Mutation testing represents a way to test a test. A common way to test the quality of quality assurance is to simulate a situation in which quality assurance should trigger an alarm. In 1971, Richard Lipton adapted this concept to testing. His idea, presented in a paper called "Fault diagnosis of computer programs," was to seed artificial defects, called mutations, into the software under test, and to check whether the test suite would find them. If the

test suite fails to detect the mutation, it would likely miss real defects, too, and thus must be improved.

MUTATION TESTING

The basic idea of mutation testing is to:

- seed lots of artificial defects into the program,
- test all defects individually,
- focus on those mutations that are not detected, and,
- improve the test suite until it finds all mutations [1].

This approach has a few benefits. First benefit is that tester can truly assess the quality of tests, not just to measure features of test execution. When modules with high risk are mutated, they can exhibit serious consequences. Third benefit comes with choice of good mutants. The more similar mutations are to real defects, the more likely you are to replicate the defect distribution in your program. Mutation is widely considered the strongest test criterion in terms of finding the most faults, but is also the most expensive.

Mutation testing is very time-consuming and as the number of mutations can easily go into the thousands, there can be several thousand build processes and test suite executions. Because of this, mutation testing requires a fully automated test [7]. There are some techniques which can improve efficiency of seeding mutants. First technique is to directly manipulate binary code. By mutating binary code rather than source code, you can eliminate the costly rebuild process after a mutation. The drawback is that binary code can be harder to analyze, in particular for complicated mutation operators. Second technique is to use mutant schemata. A mutation-testing framework produces a new mutated program version for every single mutation. However, one can also create a single version in which individual mutants are guarded by runtime conditions. Third technique is to ignore no covered code. A mutant can impact the program behavior only if it is actually executed. Therefore, programmer should mutate only statements that are covered by the test suite and run only those tests that exercise the mutation.

MUTATION TESTING GRAMMAR

Mutants can be created by mutating the grammar and then generating strings, or by mutating values during a production. Mutation can be applied to various artifacts, but it is primarily used as a program-based testing method. An input is valid if it is in the language specified by grammar, otherwise it is invalid. Any program should reject malformed inputs, which is a property that should be verified by tests. Testing could be accomplished by producing invalid strings from grammar, or by producing strings that are valid but that follow different derivation from preexisting strings. Both of these strings are called mutants.

Mutation is always based on set of mutation operators which are applied to a "ground" string. The ground string is the sequence of program statements in the program under test, and the mutants are slight syntactic variations of those statements. During program execution, the ground strings are valid inputs, and variations (mutants) are invalid inputs. For example, a valid input might be a request from a correctly logged in user in some application. The invalid version might be the same request from a user that is not logged in.

Mutation operator represents a rule that specifies syntactic variations of strings generated from grammar. Mutant represents result of one application of mutation operator over ground string. There are two issues in applying mutation operators. First is, should more than one mutation operator be applied at the same time to create one mutant? Strong empirical and theoretical evidence point that only one element should be mutated at a time. Second issue is, should every possible application of a mutation operation to a ground string be considered? Reason for this is that mutation subsumes a number of other test criteria, and if some operators are not applied, then that subsumption is lost.

When derivation is mutated to produce valid strings, the testing goal is to "kill" the mutants by causing the mutant to produce different output. Thus, mutation coverage (MC) equates to killing the mutants. The amount of coverage is usually written as percent of mutants killed and is called mutation score.

Definition for MC is: For each mutant m ∈ M, test requirement contains exactly one requirement, to kill m.

When a grammar is mutated to produce invalid strings, the testing goal is to run the mutants to see if the behavior is correct. The coverage criterion is therefore simpler, as the mutation operators are the test requirements. In this manner we have mutation operator coverage (MOC) and mutation production coverage (MPC). Definition for MOC is: For each mutation operator, test requirement contains exactly one requirement, to create a mutated string m that is derived using the mutation operator. Definition for MPC is: For each mutation operator, and each production that the operator can be applied to, test requirement contains the requirement to create a mutated string from that production

The number of test requirements for mutation depends on the syntax as well as the mutation operators. In most situations, mutation yields more test requirements than any other test criterion.

OBJECT-ORIENTED MUTATION

Object-oriented (OO) programming features changed the requirements for mutation testing. Non object-oriented mutation systems make mutations of expressions, variables and statements, but do not mutate type and component declarations. Traditional programming simply makes use of the built-in types and entities of a language which are unlikely to contain many errors. OO programs are composed of user-defined data types (classes) and references to the user-defined types. It is very likely that user-defined components contain many defects such as mutual dependency between members/classes, inconsistencies or conflicts between the components developed by different programmers.

The effectiveness of mutation testing heavily depends on the types of faults the mutation system is intended to represent. Class Mutation is a mutation technique for OO programs which particularly targets plausible faults that are likely to occur due to features in OO programming [4][8]:

- polymorphism
- method overloading
- inheritance
- information hiding
- static/dynamic states of objects
- exception handling

A. Polymorphism

In object-oriented systems, it is common for a variable to have polymorphic types. Polymorphism represents a property that a variable at runtime may refer to an object of a different type. This raises the possibility that not all objects that become attached to the same variable correctly support the same set of features. This may also cause runtime type errors which cannot always be detected at compile time [3] [5]. Two mutation operators, CRT (Compatible Reference Type replacement) and ICE (class Instance Creation Expression changes), were designed to address this feature

CRT operator replaces a reference type with compatible types. For instance, the class type S can be replaced with the class type T provided that S is a subclass of T, or S can be replaced with the interface type K provided that S implements K.

The original code:

```
S s = new S();
```

CRT mutants:

```
T s = new S();
K s = new S();
```

ICE operator is designed to change the runtime type of an object. This results in calling the constructors of compatible types, which will create the objects of the replaced types.

The original code:

```
S s = new S();
```

ICE mutant:

```
S s = new T();
```

B. Method Overloading

A class type may have more than one method with the same name as long as they have different

signatures. When several versions of the same name method are available, there is a great possibility to an unintended method be called [6]. Method overloading feature can be handled by manipulating parameters in method declarations and arguments in method invocation expressions. Four mutation operators, POC (method Parameter Order Change), VMR (oVerloading Method declaration Removal), AOC (Argument Order Change), and AND (Argument Number Decrease), were designed to address this feature.

The POC operator changes the order of parameters in method declarations if the method has more than one parameter.

```
The original code:
    public LogMsg(int level,
                  String logKey,
                  Object [] inserts) {…}
    public LogMsg(int level,
                  String logKey,
                  Object insert)  {…}

POC mutant:
    public LogMsg(String logKey,
                  int level,
                  Object []inserts) {…}
```

In the example, the POC operator creates a mutant of the first constructor by swapping the first and second parameters of the constructor. This mutant program is executed without compilation errors in spite of the fact that the types of the swapped parameters are totally different. The reason is that the instance creation expressions that call the first constructor in the original code are directed to the second constructor when the mutant is executed, because the second constructor is now better fitted. It is possible because arrays can be assigned to variables of type Object in Java. This example shows that there is a possibility of invoking a wrong constructor/method among the overloaded constructors/methods due to an unintended parameter type conversion.

VMR operator removes a whole method declaration of overloading/overloaded methods. In this way, tester can check whether the right method is invoked for the right method invocation expressions. The VMR operator can also provide coverage for the method overloading feature, i.e., checking if all the overloading/overloaded methods are invoked at least once – because test data must reference the method in order to notice that that method has been deleted.

AOC operator changes the order of arguments in method invocation expressions, if there is more than one argument. For example, the following mutant produced by the AOC operator represents the error of a wrong argument order and as both arguments have the same type (Java String), the order change in the mutant did not cause compilation problems.

```
The original code:
    Trace.entry("Logger", "addLogCatalogue");

AOC mutant:
    Trace.entry ("addLogCatalogue", "Logger");
```

AND operator reduces the number of arguments one by one, if there is more than one argument in method invocation expressions. The original code has two different trace methods:

```
public void trace(int level, Object obj, String text)
{…}
public void trace(Object obj, String text) {…}

The original code:
    Trace.trace(Trace.Event,this,sccsid);

AND mutant:
    Trace.trace(this,sccsid);
```

Although the mutant has two arguments instead of three, it is successfully compiled because class Trace has two different trace methods. The original code calls the first method while the mutant calls the second method.

C. Inheritance

A class type may contain a method with the same name and the same signatures as the method declared in superclasses or superinterfaces. In this

case, the method in a subclass overrides the method of a superclass (method overriding/hiding). When there is more than one method of the same name, it is important for testers to ensure that a method invocation expression actually invokes the intended method [2]. OMR (Overriding Method Removal) operator is designed to check that overriding/overridden methods are invoked appropriately.

OMR operator removes a declaration of an overriding/hiding method in a subclass so that a reference to the overriding method goes to the overridden/hidden method instead. If a test set fails to see any difference whether the overriding method is called or the overridden method is called, it implies that the current test set is inadequate. The OMR operator also checks coverage for the method overriding feature – i.e., overriding and overridden methods are invoked at least once.

A Java class may have two or more fields with the same simple name if they are declared in different interfaces and/or in classes. In this case, the field variables defined in a class hide the fields of the same name declared in superclasses or superinterfaces. While this feature is powerful and convenient, it might cause an unintended field being accessed, especially in a long and complex class hierarchy. The intent of the HFR (Hiding Field variable Removal) and HFA (Hiding Field variable Addition) operators is to check that hiding and hidden fields are accessed appropriately

HFR operator removes a declaration of a hiding field variable so the references to that field actually access the field in a superclass or a superinterface. This operator ensures that a test set is able to distinguish referencing a hidden field from referencing a hiding field. If a test set produces the same output even if a hiding field is removed, it indicates the test set is inadequate.

HFA operator adds field variables that appear in superclasses/ superinterfaces into the class under mutation so that the added fields hide those in superclasses/ superinterfaces.

Both the HFR and HFA operators check the coverage of field variables in the presence of inheritance because test data must accesses the hiding/hidden and inherited fields at least once. The difference is that the HFA operator checks that inherited fields are accessed at least once whereas the HFR operator checks that hiding/hidden fields are accessed.

D. Information Hiding

Object-oriented languages provide an access control mechanism that restricts the accessibility/visibility of attribute variables and methods. It is an important testing role to make sure that a certain access mode provides and restricts its intended accessibility/visibility at all times. The intended access control can also be broken in connection with other OO features such as inheritance. Java provides four possible access modes: public, private, protected, and default.

AMC (Access Modifier Changes) operator manipulates Java access specifiers to address the information hiding feature. This operator replaces a certain Java access mode with three other alternatives. The role of the AMC operator is to guide testers to generate enough test cases for testing accessibility/visibility. For example, a field declaration with a protected access mode will have three mutants.

The original code:

```
protected Address address;
```

AMC Mutants:

```
public Address address;
private Address address;
Address address; //default
```

E. Static/Dynamic States of Objects

Java has two kinds of variables – class and instance variables. The Java runtime system creates one copy of each instance variable whenever an instance of a class is created (dynamic). Class variables are allocated once per class, the first time it encounters the class (static).

SMC (Static Modifier Changes) operator is used to examine possible flaws in static/dynamic states. The SMC operator removes the "static" modifier to change a class variable to an instance variable or adds

the modifier to change an instance variable to a class variable.

The original code:

```
public static int VALUE = 100;
private String s;
```

SMC Mutants:

```
public int VALUE = 100; //static is removed
private static String s; //static is added
```

F. Exception Handling

The most obvious mistake in exception handling is not specifying appropriate exception handlers in the required place. In Java, programmer either handles an exception (i.e., catches the exception by declaring a try catches block) or propagates it (i.e., declares it to throw in a throws statement of a method declaration). EHR (Exception Handler Removal) and EHC (Exception Handling Change) operators are declared for the feature of exception handling.

EHR operator modifies the declared exception handling statement (try-catch-finally) in two different ways.

- it removes exception handlers (catch clause) one by one when there is more than one handler
- it removes the exception handler and finally clause in turn when there exist one handler and the finally clause

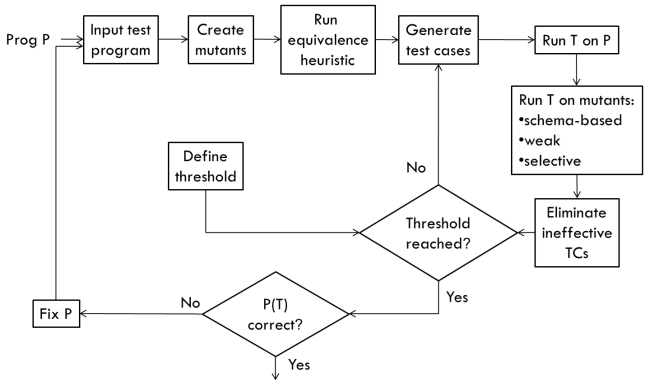
The EHR operator is not applied when there is only one handler without finally clause because it simply causes compilation errors. This operator gives coverage of catch and finally clauses.

EHC operator swaps the way of handling an exception. It catches the exception that is supposed to be propagated by changing a throws declaration to a try-catch statement, or propagates the exception that is supposed be caught within the method by changing a try-catch statement to a throws declaration.

TESTING PROGRAMS WITH MUTATION

Procedure of testing programs with mutation is shown in figure 1.

FIGURE 1 - TESTING PROGRAMS WITH MUTATION



The tester submits the program which should be tested. Automated system starts changing original statements by creating mutants. Optionally, those mutants are then analyzed by a heuristic that detects and eliminates as many equivalent mutants as possible. A set of test cases is then generated automatically and executed against the original program, and then against the program that contains mutants [9]. If the output of a mutant program differs from the original (correct) output, the mutant is marked as being dead and is considered to have been strongly killed by that test case. Dead mutants are not executed against subsequent test cases. Test cases that do not strongly kill at least one mutant are considered to be "ineffective" and eliminated. Once all test cases have been executed, coverage is computed as a mutation score (ratio of dead mutants over the total number of non-equivalent mutants). Mutation score of 1.00 means that all mutants have been detected.

A mutation score of 1.00 is usually impractical, so the tester defines a "threshold" value, which is a minimum acceptable mutation score. If the threshold has not been reached, then the process is repeated, each time generating test cases to target live mutants, until the threshold mutation score is reached. Up to this point, the process has been entirely automatic. To finish testing, the tester will examine expected output of the effective test cases, and fix the program if any faults are found. This leads to the fundamental premise of mutation testing: In practice, if the software contains a fault, there will usually be a set of mutants that can only be killed by a test case that also detects that fault.

Mutation Testing Tools

Mutation testing requires automated testing tools, which is not a trivial tool to make. Automated mutation tools must be able to parse the program and know its language. When the program is run, mutant can be killed by one of two possible scenarios:

- if a mutant crashes
- if the mutant goes into an infinite loop

The runtime system must handle both of these situations.

There are four ways to build mutation tools:

- interpretation approach
- separate compilation approach
- schema-based mutation
- reflection

A. Interpretation approach

A program under test is first parsed into an intermediate form. This is usually a special-purpose language designed specifically to support mutation. This language can easily handle the bookkeeping when mutants are killed as well as program failure. The usual way to handle infinite loops is first to run a test case on the original program, count the number of intermediate instructions executed, then run the test case on a mutant. If the mutant uses X times more intermediate instructions (X has usually been set at 10), then the mutant is assumed to be in an infinite loop and marked dead. The mutation testing tool directly modifies this intermediate form which represents a special purpose language.

This approach has several advantages:

- can easily handle the bookkeeping when mutants are killed
- can respond to program failure and infinite loops
- full control of the execution
- parsing the program and creating mutants is efficient
- creating mutants by making small changes to the intermediate form is simple
- only the rules for changing the intermediate form need to be stored on disk

Disadvantages of this approach are:

- mutation system must be a complete language system: parser, interpreter, and run-time execution engine
- complicated to implement and represents a significant investment
- slow execution (10 times slower than a compiled program). Researchers have found that it can take up to 30 minutes to run all mutants on a 30 line program.

B. Separate compilation approach

In this approach each mutant is created as a complete program by modifying the source of the original program which is under test. Then each mutant is compiled, linked and run.

Main advantage of this approach is fast execution.

However, there are several disadvantages:

- difficulties with bookkeeping when mutants are killed
- difficulties with handling run-time failures and infinite loops
- compilation bottleneck, particularly with large programs, but also with small programs that run very quickly, because the time to compile and link can be much greater than the time to execute
- difficulties with applying weak mutation

C. Schema-based approach

Schema-based approach consists of following steps:

- MSG (Mutant Schema Generation) encodes all mutations into one source-level program, called a metamutant
- metamutant is compiled and executed in the same environment at compiled program speed

These mutation systems are less complex and easier to build then interpretive systems because they do not need to provide the entire run-time semantics and environment. A mutant schema has two components,

a metamutant and a metamethod set, both of which are represented by syntactically valid constructs.

In MSG, a program schema represents a template. A partially interpreted program schema syntactically resembles a program, but contains free identifiers that are called abstract entities. The abstract entities appear in place of some program variables, data type identifiers, constants, and program statements. A schema can be instantiated to form a complete program by providing appropriate substitutions for the abstract entities.

D. Reflection

Reflection represents an approach that combines the interpretive and compiler-based approach. Reflection allows a program to access its internal structure and behavior, and manipulate that structure, thereby modifying its behavior based on rules supplied by another program.

Reflection is possible only in languages that support it (Java and C#). Both support reflection by allowing access to the intermediate form (Java bytecode). Reflection can be achieved in three ways:

- Compile-time reflection allows changes to be made when the program is compiled
- Load-time reflection allows changes to be made when the program is loaded into the execution system (JVM)
- Run-time reflection allows changes to be made when the program is executed

Reflection has several advantages:

- it allows programmers extract information about a class
- it provides an API to modify the behavior of a program during execution
- it allows objects to be instantiated and methods to be invoked dynamically
- some of the OO operators cannot be implemented via MSG

CONCLUSION

First paper about mutation testing was published 30 years ago. Only now mutation testing becomes widely implemented. The reason for this is that automated testing is much more widespread than it was 10 years ago, and there is no mutation testing without it. Computing power keeps on increasing, and we can begin to afford the huge computing requirements imposed by mutation testing. Modern test case generators make it fairly easy to obtain a high coverage automatically but still, the test cases are not good enough. There is a variety of dynamic and static optimizations that make mutation testing reasonably efficient and also highly effective when it comes to improving test suites. All this implies that mutation testing will become much more commonplace in the future.

REFERENCES

[1] Ammann P. and Offutt J., (2008) "Introduction to Software Testing", Cambridge University Press

[2] Brahma S. Punganti A., Pattanaik P.K., Prasad S. and Mall R., (2010) "Model-Based Mutation Testing of Object-Oriented Programs", Proceedings of 2nd international Conference on IT & Business Intelligence, India

[3] Finkbine R., (2003) "Usage of Mutation Testing as a Measure of Test Suite Robustness", Digital Avionics Systems Conference

[4] Ma Y.S., Harrold M.J. and Kwon Y.R., (2006) "Evaluation of Mutation Testing for Object-Oriented Programs", 28th International Conference on Software Engineering, China

[5] Ma Y.S., Offutt J., (2005) "Description of Class Mutation Operators for Java"

[6] Ma Y.S., Offutt J., (2005) "Description of Method-level Mutation Operators for Java"

[7] Riley T. and Goucher A., (2009) "Beautiful Testing – Leading Professionals Reveal How They Improve Software", O'Reilly

[8] Sunwoo K., Clark J. , McDermid J., (2000) "Class Mutation: Mutation Testing for Object-Oriented Programs", OOSS: Object-Oriented Software Systems

[9] Umar M., (2006) "An Evaluation of Mutation Operators for Equivalent Mutants", Department of Computer Science King's College, London

Submitted: November 07, 2011
Accepted: December 31, 2011

Social Media in Marketing and PR

Velimir Štavljanin, Vinka Filipović, Milica Kostić Stanković

velimirs@fon.bg.ac.rs, vinka@fon.bg.ac.rs, milicak@fon.bg.ac.rs

Faculty of Organizational Sciences, University of Belgrade

Contribution to the State of the Art

UDC **32.019.5:658.8**

Abstract: Social media as a new communication channel has managed to radicalize the way companies communicate with consumers and other stakeholders. Companies that are not on time engaged in social media weaken its ability for competitive struggle. In this paper we present possibilities of different types of social media in relation to marketing and public relations. Also, the paper will give specific recommendations for the use of social media in mark eting and public relations.

Keywords: Marketing, Public Relations, Social Media

Introduction

Historically, companies have been communicated to the public by placing certain predefined message, in order to generate planned response. Even if in the process of creating messages research was done regularly, the role of the public remained largely passive. During the 60s academic circles made some changes. These changes were intensified through the 90s thanks to the commercialization of the Internet. The changes were related to the possibility of faster information, constant availability of information, access from anywhere using not only desktops, but laptops and mobile devices and lower cost of placing information. But, in the early 21st century radical changes occurred in the use of available web technologies, because of the inflexibility of the earlier models. In past, Internet pages were able to change only by those with specific knowledge in web coding, and users were mostly readers with the minimal ability to change anything. Technologies, such as XML, AJAX, and RSS facilitated development of a wide range of new applications. Using these applications users are now able to create content on the Internet, to interact with each other, and with companies. This new era of the Internet in which it became more "human" is popularly referred to as Web 2.0. Term Web 2.0 was coined in 2004 at the conference on these new Web technologies. One of the Web 2.0 definitions says that it is a new platform which should take effects and the collective intelligence of the network as a basis for building applications that will attract users (O'Reilly). Nowadays, as the nature of the Internet has become interactive, habits have changed, and the way we consume it (Ryan & Calvin, 2009). The conversation has become a common form of participation, regardless of geographical, temporal and cultural boundaries. In order to hear the voice of consumers companies had to change their habits on the Internet and begin to participate in that conversation. However, many companies were making a mistake by engaging in a conversation without the knowledge of social media and new principles that govern the social media. In order to achieve success in conversation, they must first listen to the environment, and to build presence on the social media.

Social media

Many associate social media with well-known sites such as Facebook, YouTube, MySpace, Twitter, are intended for general interactions. But today there is a trend and already hundreds of social media sites that focus on smaller groups with specific interests

(Frick, 2010). Due to its diverse nature social media is not easy to define. One simple definition is that social media is a collection of Web pages and applications that were developed to allow users to interact with their friends (Brown, 2010). Social media can be defined as online tools and platforms that people use each other to exchange views, ideas, experiences and perspectives (Lincoln, 2009). Social media is the unifying term for software and services based on the Web that allows users to socialize online, exchange, discuss, communicate and participate in all forms of social interaction (Ryan & Calvin, 2009). The interaction can include text, audio, video and other media, individually or in combination. In addition, social media allows content generation, sharing of existing facilities, reviews and evaluation, discussion on issues of concern, sharing experience and expertise – i.e. all that can be shared and distributed via digital channels. A number of sites now include elements of social media to engage audiences, and some sites build their business model around social media, user participation and user generated content.

Different forms of social media

Social media sites are based on different models, but with the same premises of interaction, creation, exchange and sharing of content, content evaluation and discussion. Content is dynamic and can be linked to site, individual article, blog or blog post, photo, audio or video material, question or a comment of another user, i.e. with anything that can be distributed in digital form. Most social media sites can be identified within one category, but they are more often a combination of several social components. However some basic characteristics are used for their classification in categories. Categorization differs from author to author and may include sometimes a large number of categories. Savko presented a comprehensive list that contains 15 categories of social media (Safko, 2010):

- Social Networking
- Publishing
- Photo sharing
- Audio
- Video
- Microblogging
- Livecasting
- Virtual worlds
- Gaming
- Productivity applications
- Aggregators
- RSS
- Search
- Mobile
- Interpersonal

Focus of Lincoln classification is on the most important applications. 10 key tools of social media are (Lincoln, 2009):

- Blogging
- Microblogging
- RSS
- Widgets
- Social Networks
- Chat rooms
- Message Boards
- Podcasts
- Video sharing
- Photo sharing

Social Networks

Social networks are applications that continue the basic idea of the Internet. The predecessor of the Internet, ARPANET, was created in order to facilitate networking between universities. Some of the first applications Usenet, LISTSERV and BBS possessed many characteristics that have social networks. Today's social networks can be formed around common interests, attitudes, views, family life, religious beliefs, race or other similarities. Basically, social networking sites allow construction of a network of "friends" with whom users can share a multitude of digital resources. Sites have ability to search and connect with other profiles, instant communication, sharing content and files. Social networks gather hundreds of millions of people. Only Facebook has over 750 million profiles. Besides Facebook, the most popular global networks are already mentioned MySpace, LinkedIn, and Ning, Google +, orkut, hi5, bebo. Profiles can be individuals or groups, but also companies, specifically corporate brands and their brands of products. The main benefits of social networks are increased visibility and impact on reputation. Companies can communicate with individuals, whose activities can

be monitored in order to reveal more details about them and deliver content of value. Such content users will share and expand its range. If the company is constantly present, if the content is always up to date, relevant and of value to users, and if it provides feedback to users, it will affect the reputation positively. In addition, social networks are used to identify and attract individuals who are active on the network and which can act as advocates or evangelists of the company. Planning a social network presence is a risky activity. As noted above, social networks can be useful if it offers content of value to users. However, if company just wants to promote itself, it will not be rated by users as valuable, and this approach can cause adverse reactions or the tide of negative feedback in an open environment that can get out of control and create a crisis.

Blog

Blog is one of the first forms of social media that has become popular as a medium for communication and personal presentation. . The word blog is derived from two words - web and log. Blogs are collection of records shown in reverse order. Great impact blog had on the Internet population is reflected in a shift that has been made from the web that is static, and where the users are only readers, to interactive web where the users are those who publish and those who are in dialogue. Today many blogging platforms, open-source and commercial, allow users to easily create blogs, and even entire sites that are based on blogs, like WordPress.

Blogs are written on a regular basis, sometimes daily, sometimes weekly or monthly. Bloggers are not just individuals who offer opinions from a personal point of view. Blogs are often maintained by individuals from companies that write from a personal point of view (blog Mini-Microsoft maintained by anonymous Microsoft employee) or on behalf of the corporate brand (Bill Marriott chairman and chief executive of Marriott International blog), or product brand (Opel employee maintained blog about brand Meriva - www.meriva-blog.de). If bloggers are individuals from the company, it is always shown clearly who is responsible for the blog. Blogs have their regular visitors who read more or less often posts. Posts can be commented or even carried on other blogs or other social networks. This creates a viral effect and increases the visibility of a brand, and that is very important. Blog simplicity, a belief in that is authentic, honest and of the authority provides great visibility and public involvement.

Microblogging

Microblogging is relatively new form of social media. Microblogging is a medium in the form of blogging, a sort of short text blogging. Similar to SMS on the mobile phones, goal of microblogging is to ensure fast and timely notification. Message length is limited to 140 characters. This limitation allows the possibility that message can be transmitted not only by using desktop application, but also using mobile applications and even SMS. Although similar in form as blogging, nature of microblogging use is completely different. Microblogging is used by companies primarily as an information tool. It is of great importance for the realization of the events. A leader in the microblogging field Twitter, is also a pioneer. Other popular microblogging platforms are Jaiku and Pownce. Twitter was launched in March 2006 as a result of a research project conducted by a small company Obvious from San Francisco. Initially Twitter was used for internal communication among employees. In October 2006 Twitter was launched to the public.

Value of microblogging is not only in monitoring individual posts, rather the aggregation of multiple sources from the same area and a quick overview of the state. Some of the microblogging roles are the opinion poll, by listening opinion leaders and their followers and fast communication. Microblogging is very useful in the integrated appearance when companies need to generate site traffic or raise the level of interest.

Wiki

Wikis are online collections of web pages that are open for anyone to create, edit, discuss, generally to contribute. The first wiki was the WikiWikiWeb site created by Howard Cunningham in 1994. Name Wiki originates from the Hawaiian word for quick. The best known example is Wikipedia, launched in January 2001. During the first year Wikipedia gen-

erated over 20,000 articles in 18 languages. Today Wikipedia has over 3.7 million articles on English. It has long been criticized for the accuracy and authorship of articles, remains as one of the most visited sites. What makes wiki a tool of choice is a simple community creation consisting of people who cooperate by sharing their knowledge, experience and expertise online. In this type of community articles constantly evolve during time. Their relevance is higher as the time goes by and as community grows. There are many examples where the wikis are used as internal communication portals, or as the external communication tools for brand community building.

Social Bookmarking

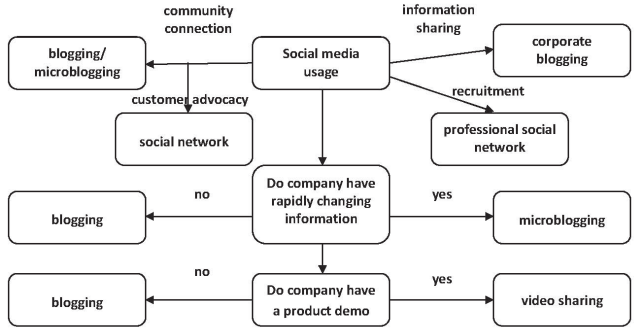
Social Bookmarking is a favorite way of organizing, storing and managing resources on the Web 2.0. Social bookmarking sites such as Delicious, Ma.gnolia, StumbleUpon, Digg and others allow users to record bookmarks for their favorite web resource (page, audio, video or whatever) and categorize them by using the tags (which can be predefined in system or defined by the user). The procedure is similar to Favorites adding in browser. Resources can be then sorted in chronological order or by categories or by tags. In such open systems, it is possible to bookmark favorite resources as private or public. As a public it will be available to all users and the social bookmarking system and it can be categorized later even with tags from other users. The specificity of these sites is search, which is different from the results that offer classical search engines, based on human intuition.

In this way the favorite content bookmarks stored on the Internet could be accessed from anywhere at any time and from any device that has Internet access. This content is much easier to search and share. For companies, such sites are particularly important because of the possibility to increase their visibility and to provide user tags which will make it easier to search, but also affect the relevance and authority.

The choice of social media

Companies usually prefer a combination of social media, because of their specific target audience. The decision which social media will be used depends on several factors. The author Brown presented one of the possible forms of choosing the right social media to achieve certain goals (Brown, 2010). The choice flowchart is shown in figure 1.

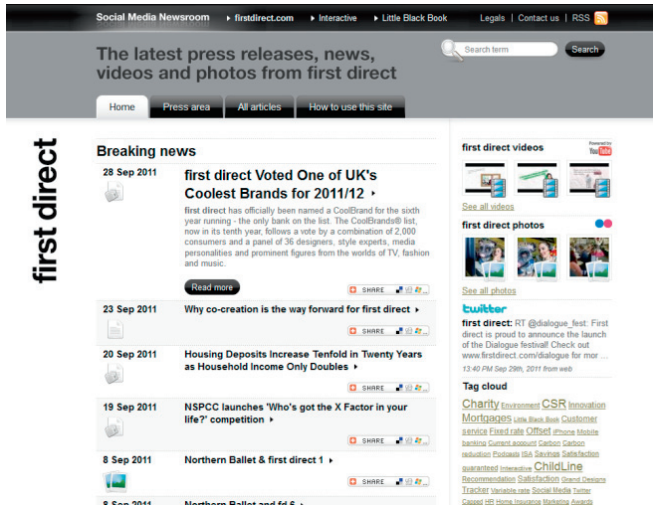
Figure 1. Social media tools flowchart based on Brown



Social Media Release

Social Media Release (SMR) is viewed as an addition, and somehow as a replacement for traditional press release. Simple explanation is that the social press release (SMR) is a press release that is published on the internet and done in such a way that its content is easily distributed.

Figure 2 SMR Example



Although it appeared different from standards for the SMR, there are common aspects that distinguish them from traditional press releases placed on the Internet. It is important to note that the SMR cannot be sent via e-mail journalists or bloggers. SMR is something which reveals or calls to see. SMR has links to variety of social media, social networks, bookmarking, tag

search and links to other relevant content. Tags in SMR allow easier search and documents discovery. Through links to bookmarking sites, it can be easily distributed and monitor on RSS readers. SMR may include multimedia content, in the form of pictures or videos.

Brian Solis, one of the leading experts in PR 2.0, summarized the content of SMR, in the following list (Brown, 2009):

- headline;
- intro paragraph, including keywords;
- supporting facts;
- selection of quotes;
- multimedia – audio, video and images;
- RSS - company and/or product news;
- link to insert in social networks (Facebook, Bebo, MySpace, hi5 or others);
- blog this (link to blogger platforms);
- share on Twitter, Jaiku, Pownce or Tumblr;
- other bookmarks;
- other relevant links;
- links to news aggregators and communities including Digg and reddit;
- further information details and links could include an image plus vCard, or links to LinkedIn, Facebook or Twitter feeds.

SMR is still in its developmental stage, and there are those for and against such a solution. Some of the advantages are that they are fully electronic, that they can be easily detected through defined tags and links to various social networks. In relation to the time in which it was critical to send a little larger picture, it is now possible to watch online movies in HD, which is affirmative for the SMR.

Social Media Newsroom

Social Media Newsroom (SMN) is primarily designed as a place where the Social Media Release (SMR) will be published. Of course this is not the exclusive place for SMR publishing, rather a solution that fits into the concept of Web 2.0. SMN is the evolution of digital media sites, which have already been present for years as a part of corporate web sites. Traditional media sites acted mainly as an archive of press releases, photos and videos, but the purpose of the SMNx is to encourage sharing and dialogue.

SMN contains many features of traditional media sites such as different types of content, press releases, reports and pictures. On the SMN press and other, stakeholders can find information about top management, including photos and bios. SMN may contain press release and photo archives. One of the features of SMN is corporate calendar with dates for key announcements and key events. Besides all the traditional content, SMN will include more interactive features, as well as links to specific topics, which would be sent by e-mail or distributed through an aggregator. SMN can contain a multimedia library in addition to conventional photo library. In addition, there would be a section with a choice of RSS and links to sites for social bookmarking. SMN can provide a direct conversation about information or specific statements on the company's website.

Figure 3 SMN example

Conclusion

Social media opened new opportunities for the marketer. In the same time it is very risky to participate without sound planning. These risks are related to the definition of social media and facts that social media is a media where users exchange views, understandings, experiences and perspectives in an open environment. In order to properly plan their campaigns, managers should first familiarize with the new media, its advantages and disadvantages and opportunities for participation. This paper can be useful for marketing and public relations managers as a basis for successful participation in social media.

References:

[1] Bernal, J. (2009), Web 2.0 and Social Networking for the Enterprise, IBM Press

[2] Brown, E. (2010), Social Media Marketing for Business, BSC - The Chartered Institute for IT

[3] Brown, R. (2009), Public relations and the social web: using social media and Web 2.0 in Communications, Kogan Page Limited, London

[4] Frick, T. (2010), Return on Engagement: content, strategy, and design techniques for digital marketing, Focal Press

[5] Lincoln, R. S. (2009), Mastering Web 2.0, Kogan Page Limited, London

[6] Phillips, D. and Young P (2009), Online public relations: a practical guide to developing an online strategy in the world of social media, 2nd ed., Kogan Page Limited, London

[7] Ryan, D. and Calvin J. (2009), Understanding Digital Marketing: Marketing strategies for engaging the digital generation, Kogan Page Limited, London

[8] Safko, L. (2010), The Social Media Bible: Tactics, Tools, and Strategies for Business Success, 2nd ed, John Willey & Sons, Hoboken, HJ

[9] Shah, R. (2010), Social Networking for Business: Choosing the right tools and resources to fit your needs, Wharton School Publishing

[10] O'Reilly, T., "What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software" article from the website http://oreilly.com/pub/a/web2/archive/what-is-web- 20.html? page = 1, accessed on 01.10.2011

Submitted: December 22, 2011
Accepted: December 28, 2011

Comparative Implementation Analysis of AES Algorithm

Boris Damjanović[1], Dejan Simić[2]
[1]*dboris0206@gmail.com,* [2]*dsimic@fon.bg.ac.rs*
Faculty of Organizational Sciences, 11000 Belgrade, Serbia

Case study
UDC **659.2:004.651**

Abstract: Advanced Encryption Standard (AES) is the first cryptographic standard aroused as a result of public competition that was established by U.S. National Institute of Standards and Technology. Standard can theoretically be divided into three cryptographic algorithms: AES-128, AES-192 and AES-256. This paper represents a study which compares performance of well known cryptographic packages - Oracle/Sun and Bouncy Castle implementations in relation to our own small and specialized implementations of AES algorithm. The paper aims to determine advantages between the two well known implementations, if any, as well as to ascertain what benefits we could derive if our own implementation was developed. Having compared the well known implementations, our evaluation results show that Bouncy Castle and Oracle/SUN gave pretty equal performance results - Bouncy Castle has produced slightly better results than Oracle/Sun during encryption, while in decryption, the results prove that Oracle/Sun implementation has been slightly faster. It should be noted that the results presented in this study will show some advantages of our own specialized implementations related not only to algorithm speed, but also to possibilities for further analysis of the algorithm.

Keywords: computer security, cryptography, algorithms, standards, AES, performance.

Introduction

In the literature, there is a certain number of Java cryptographic APIs [10][16] (Application Programming Interfaces). Most of these implementations are constructed with an intention to, as much as possible simplify usage of various cryptographic algorithms and techniques. However, much smaller number of people is engaged in research and implementation of individual algorithms and cryptographic techniques.

Constructing small, specialized implementations of some algorithm which function is devoted to the specific task gives us multiple benefits. Such implementations generally achieve better results accomplishing the mission for which they are made for. In addition, the writing of these programs allows the author to be well acquainted with the ways of functioning of the individual algorithms and to come up with new discoveries related to the different points of view.

This paper represents an empirical study which compares performance of massive and well known cryptographic packages in relation to our own implementations [4] of AES [6][5][3][2][14] algorithm in Java programming language. In the article, we will further compare these results with speed measurements of an experiment with AES algorithm extensions below the key size of 128 bits. As a reference for measuring, we will use two AES implementations, which are parts of the large cryptographic packages - Bouncy Castle [11] and Oracle (former Sun), which both use the Java Cryptography Extension (JCE) [10][16]. Cryptographic implementations in the Sun JDK are distributed through several different providers still using name Sun ("Sun", "SunJSSE", "SunJCE", "SunRsaSign").

Known cryptographic packages and the length of the keys used in the experiments are:

1. Oracle/Sun JCE [13] version 1.7 with 128, 192 and 256 bit encryption;
2. Bouncy Castle [11] version 1.46 with 128, 192 and 256 bit encryption.

Comprised in our evaluation, we had four of our own implementations as well:

1. Implementation of AES standard algorithm (with 128, 192 and 256 bit key length), each based on Dr. Gladman's [9][12] and Bertoni's [1] ideas;
2. Implementation of the expanded AES algorithm (with key lengths of 32 and 64 bits), based on Dr. Gladman's and Bertoni's ideas each.

TEST PLATFORM

As a test platform was used an Asus notebook computer with Intel (R) Core (TM) i5 450M processor at 2.40 GHz, (without new AES set of instructions - AES-NI) with 4GB RAM and Seagate@Momentus@ ST9500325AS hard disk and with the MS Windows 7 operating system.

As a development environment we used Eclipse Java EE IDE for Web Developers, Build id: 20110916-0149, Java SE Development Kit 7u1 for Windows and Java Cryptography Extension (JCE) for Java SE Development Kit 7u1 for Windows.

IMPLEMENTATION DETAILS

In our own implementation of the AES algorithm we used POJOs (Plain Old Java Objects). In this implementation we experiment with possible extensions of this algorithm according to the simple rules that we will introduce later in the text. Because of these extensions, our own implementation will hereinafter be referred to as EAES (Expanded AES).

To determine how fast our implementation is, we will compare it with implementations of well known manufacturers that use Java Cryptography Extension (JCE) [10][16] – Oracle/SUN and Bouncy Castle [11]. Both implementations are using provider-based architecture. For more details on the implementation of various cryptographic algorithms in Java, readers are referred to [10], [11], [16].

AES algorithm described in FIPS-197 document [6] transforms 128 bit block of data during 10, 12 or 14 rounds using the initial key lengths of 128, 192 and 256 bit. The initial key is then enlarged to (10+1)*16, (12+1)*16 or (14+1)*16 bytes in the key expansion routine. Each round repeats the SubBytes(), MixColumns(), ShiftRows() and AddRoundKey() transformations. AES authors redefine both addition operation within the $GF(2^8)$, which is then conducted by XOR operation at the byte level and multiplication operation which is thus conducted as polynomial multiplication with the conditional modulo polynomial 0x11B. The mentioned multiplication is the most time consuming in the aspect of optimization, because it is intensively used during the MixColumns() transformation.

The most known software implementations of AES algorithm are based on Dr. Gladman's ideas. These implementations use four lookup tables of 4kB each for encryption, commonly referred to us as T tables, and four additional tables of same size for decryption. These tables contain the intermediate results calculated in advance for several transformations at once.

Beside the aforementioned eight large tables, we must point out two smaller tables of 256 bytes in size each, for SBox and inverse SBox, as well as a table with calculated values of RCon operation for which it is usually sufficient to allocate eleven bytes. In those implementations the 128 bit block (State) is represented as a 4x4 byte matrix, and it is processed on column by column basis.

According to Bertoni's idea, State matrix is to be firstly transposed then processed on row-by-row basis. This approach uses only three smaller tables - SBox, inverse SBox and RCon, therefore consumes significantly less memory [1], but uses multiplication more intensively.

HYPOTHESES

As mentioned fastest software implementations of AES algorithm today are based on ideas of Dr. Brian Gladman [9][12]. These implementations are characterized by the high processing speed, which is based

on pre-calculated tables, due to which a great deal of memory is used. On the other side, there is a very interesting idea of Bertoni that achieves very good performance with significant decrease in memory usage [1], because the idea is based on a significantly smaller utilization of pre-calculated tables with interim results.

To conduct the necessary experiments with higher quality, we implemented both ideas in Java programming language, so that the implementation by Dr. Brian Gladman is marked by EaesG, while slightly changed implementation of Bertoni's ideas is marked by EaesB. You may have already assumed that the letter E in the mark refers to our implementations that reduce the standard to 32 and 64 bit encryption/decryption.

Experiments to be carried out will serve to test the following hypotheses:

- Specialized implementation of AES algorithm shows equally good or better results compared to the well known cryptographic packages,
- Large cryptographic suites lose a lot of the time for the first initialization at engine startup,
- Experimental extensions of AES algorithm for 32 and 64 bit encryption and decryption are achieving even greater differences in processing speed compared to the large cryptographic packages.

Testing methodology

To achieve the highest test results precision, we implemented four applications named SunAes, BcAes, EaesB and EaesG. Each individual implementation was given the same conditions in regard to processor, memory and hard disk usage. Each particular implementation was evaluated using the same test platform as described in section 2. All tests were conducted by consecutive repetition of measurements on files in 512KB-32MB size.

The first series of tests was conducted in such manner that we measured the time required for initialization of particular class, loading data from disk, its processing and saving to disk. Then, to avoid any caching by operating system and hardware, we initiated the subsequent application in another folder, and then the following application in the third folder, etc. After that we computed the arithmetic mean of the achieved results. This way of testing showed that large cryptographic packages (such as Oracle/SUN JCE and Bouncy Castle) consume a lot of time (from 200 to even 700 ms) for initialization, while our implementation was significantly faster due to short initialization time. When we put the same code in the loop, we got significantly different results, as you can see from the following example:

```
infile_16_bytes.txt, aes128, pass: 1
 Time : 641 ms
infile_16_bytes.txt, aes128, pass: 2
 Time : 0 ms
infile_16_bytes.txt, aes128, pass: 3
 Time : 0 ms
```

Code 1: Total time results in loop

This way of testing can give us a twisted picture of large cryptographic packages speed – those are ultimately optimized and extraordinary fast implementations. However, in some applications, the extended time needed for initialization can present a problem which must be taken into account.

That is why we applied a slightly different solution in the following testing series. Firstly we slightly altered the source code, to be able to measure only the time needed for data processing. In accordance with [7][8] and [15] we conducted additional two measuring series. In the first series we measured the time by alternate starting of each application individually, to avoid the influence of caching by the operating system and hardware as much as possible. Achieved results in this step represent the arithmetical mean of five conducted measuring sessions, in which we rejected the highest and lowest result to avoid the influence of other processes in the system. In the second testing series, we put the measurement code in the loop and executed it for six times within one VM call, after which we rejected the first result, which, according to [8] is considered to be the time required for compiling. We took into account only the time required for execution. In the end, we combined two described testing methodologies as to compute the arithmetic mean of the achieved results from the last two test series. Finally, the results are presented as the mean number of milliseconds per megabyte.

Measurement Results– Standard-Defined AES Algorithm

Hereby we set out the measurement results, with the aim to rank our implementations – EaesG and EaesB in comparison to large cryptographic packages.

TABLE 1: 128 BIT ENCRYPTION RESULTS

128-bit encryption	Sun (ms)	EaesG (ms)	BC (ms)	EaesB (ms)
512 KB	4	15	11	14
4096 KB	73	84	69	93
8192 KB	144	140	158	175
16384 KB	290	312	293	365
32768 KB	591	577	593	702
ms/MB	18	18	18	22

FIGURE 1: 128 BIT ENCRYPTION RESULTS

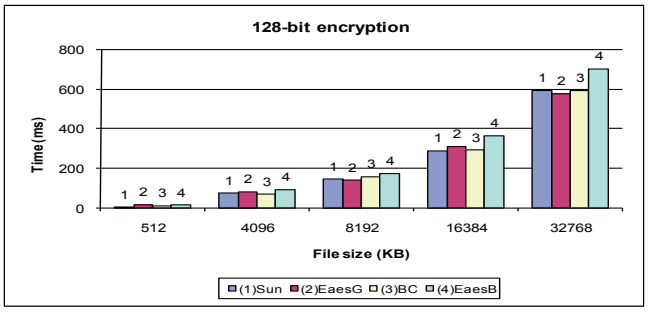
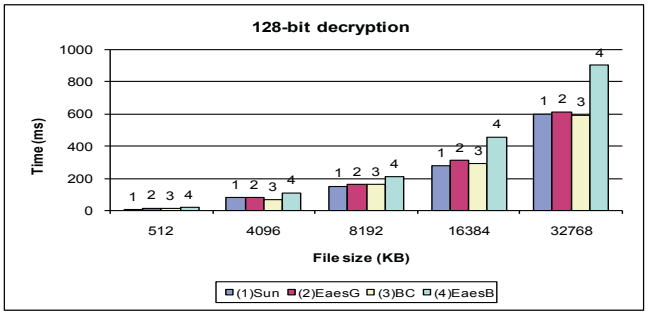


TABLE 2: 128 BIT DECRYPTION RESULTS

128-bit decryption	Sun (ms)	EaesG (ms)	BC (ms)	EaesB (ms)
512 KB	8	15	13	18
4096 KB	79	82	70	111
8192 KB	152	162	161	214
16384 KB	278	311	292	458
32768 KB	600	614	594	902
ms/MB	**18**	**19**	**18**	**27**

FIGURE 2: 128 BIT DECRYPTION RESULTS



Although all tested implementations showed impressive speed, generally speaking, our implementation based on Dr. Gladman's ideas, Bouncy Castle and Oracle/SUN implementations provided slightly better results in the described measuring conditions. Those implementations gave pretty equal results in measuring of 192 bit and 256 bit encryption and decryption:

TABLE 3: 192 BIT ENCRYPTION RESULTS

192-bit encryption	Sun (ms)	EaesG (ms)	BC (ms)	EaesB (ms)
512 KB	11	16	10	15
4096 KB	94	88	89	105
8192 KB	193	182	166	210
16384 KB	375	364	351	403
32768 KB	721	671	688	846
ms/MB	**22**	**21**	**21**	**25**

FIGURE 3: 192 BIT ENCRYPTION RESULTS

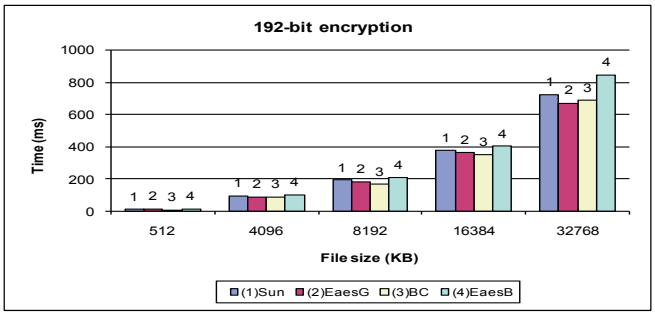
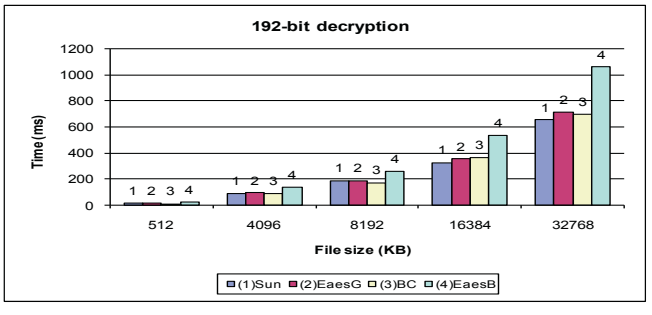


TABLE 4: 192 BIT DECRYPTION RESULTS

192-bit decryption	Sun (ms)	EaesG (ms)	BC (ms)	EaesB (ms)
512 KB	16	14	11	21
4096 KB	88	94	93	139
8192 KB	188	186	171	262
16384 KB	325	354	364	534
32768 KB	653	714	698	1060
ms/MB	**21**	**22**	**22**	**33**

FIGURE 4: 192 BIT DECRYPTION RESULTS



Once again, our EaesG and Bouncy Castle implementations encrypt data slightly faster than implementation based on Bertoni's idea. We had the least available information on Bertoni's idea, according to [1] probably for Bertoni's work had been under patenting process. We therefore gave up making any attempts to optimize implementation based on his idea. Yet, it was included in our test, because we believe that it was an awesome idea with enormous potential for experiments on standard-defined AES algorithm expansion.

TABLE 5: 256 BIT ENCRYPTION RESULTS

256-bit encryption	Sun (ms)	EaesG (ms)	BC (ms)	EaesB (ms)
512 KB	9	15	15	18
4096 KB	110	114	107	136
8192 KB	186	197	209	241
16384 KB	436	394	414	504
32768 KB	835	852	789	998
ms/MB	25	25	25	31

FIGURE 5: 256 BIT ENCRYPTION RESULTS

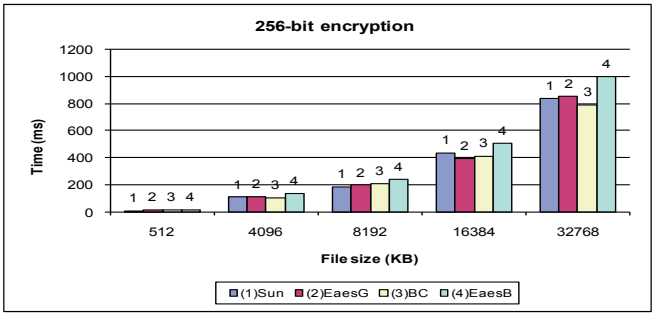
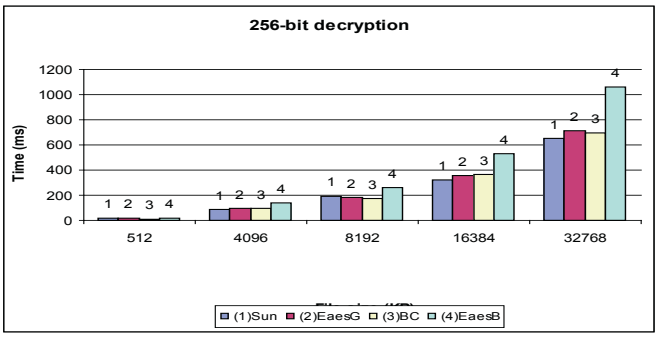


TABLE 6: 256 BIT DECRYPTION RESULTS

256-bit decryption	Sun (ms)	EaesG (ms)	BC (ms)	EaesB (ms)
512 KB	23	17	21	22
4096 KB	91	103	108	149
8192 KB	187	193	214	306
16384 KB	393	405	409	621
32768 KB	770	805	826	1234
ms/MB	**24**	**25**	**25**	**38**

FIGURE 6: 192 BIT DECRYPTION RESULTS

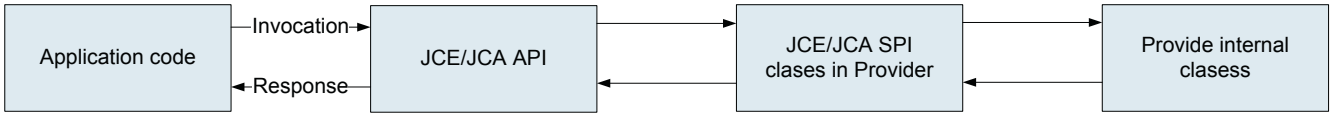


As we come to 256-bit encryption, all three implementations showed equally good results, but for the 256-bit decryption, SUN's implementation has produced slightly better outcomes, to EaesG and BC implementations respectively. Hereby we must stress out that the purpose of the described tests was not to run a dispute over the speeds of large cryptographic packages. If we exclude time needed for initialization, for the shown differences in speed are still insignificant. The complete initial test phase was conducted in order to create a solid ground for determining the real gains in speed expected to attain in our subsequent experimental implementation of 32 and 64 bit expansion of AES algorithm.

EXPERIMENT - DETAILS OF EXPANDED ALGORITHM IMPLEMENTATION

The development of one's own implementation of some cryptographic algorithm makes the essential advantage as a possibility for further specialization in certain applications. It is noticeable that a short and specialized implementation of AES algorithm produces equally satisfactory results and even faster than the ones in large multipurpose implementations. Tested Oracle/SUN JCE and Bouncy Castle implementations use "provider based" architecture, as it is shown in Figure 6a. Objects that provide functionality in the Java Cryptography Architecture (JCA) and its successor Java Cryptography Extension (JCE) are not visible to those who develop an application. Developer in the case of JCA and JCE address to collections of classes that serve as links that provide some cryptographic service. Therefore, the mentioned multifunctional implementations need more time for initialization of proper algorithm, and thus for the execution.

FIGURE 6A: PROVIDER BASED ARCHITECTURE



However, performance gain is not the only benefit of writing your own implementation of a particular algorithm. A lot more than mere speed is gained by acquiring the knowledge needed for mastering the certain algorithm – knowledge that can be used for certain improvements of this algorithm. In further text we will present two experiments that explore the possible ways for expansion of AES algorithm, related to 64-bit and 32-bit encryption.

We have already mentioned that the standard-defined AES algorithm transforms the data during 10, 12 or 14 rounds and that the initial key in the key expansion routine is developed at (10+1)*16, (12+1)*16 or (14+1)*16 bytes. Hence, AES uses 10 rounds for the 128-bit encryption, and the initial key is expanded to (10+1)*16=176 bytes. If we continue to follow this logic, for the 64-bit encryption we can use 8 rounds, due to which we will expand the initial key to (8+1)*16=144 bytes, while for the 32-bit encryption we will use 7 rounds, and the initial key will be expanded to (7+1)*16=128 bytes.

This reduction in the number of operations (via the reduction in the number of rounds) should result in certain accelerations, which we must determine by new series of tests.

Measuring Results – Expanded AES Algorithm

Based on the previously conducted measuring sessions we have ranked our implementations in comparison to well known cryptographic packages. The purpose of conducting the following series of tests was to determine the time spared by applying 64-bit and 32 bit encryption in relation to 256, 192 and 128-bit encryption and decryption. For this measuring series we also used the formerly described combination of two testing methodologies to get more precise results, and all the measurements were conducted on both of our implementations (EaesG and EaesB).

The above diagrams show the results of measurements the EaesG algorithm based on Dr. Gladman's ideas, which are marked 1 to 5, while the results of measuring the EaesB algorithm, based on Bertoni's ideas are presented with bars 6 to 10. If we observe each implementation individually, the achieved re-

TABLE 7: 256, 192. 128 BIT VS. 64/32 BIT ENCRYPTION RESULTS

Encryption	EaesG 256	EaesG 192	EaesG 128	EaesG 64	EaesG 32	EaesB 256	EaesB 192	EaesB 128	EaesB 64	EaesB 32
512 KB	15	16	15	16	16	18	15	14	12	16
4096 KB	114	88	84	63	63	136	105	93	79	55
8192 KB	197	182	140	103	94	241	210	175	158	139
16384 KB	394	364	312	270	224	504	403	365	308	271
32768 KB	852	671	577	484	442	998	846	702	608	529

TABLE 8: 256, 192, 128 BIT VS. 64/32 BIT DECRYPTION RESULTS

Decryption	EaesG 256	EaesG 192	EaesG 128	EaesG 64	EaesG 32	EaesB 256	EaesB 192	EaesB 128	EaesB 64	EaesB 32
512 KB	17	14	15	12	7	22	21	18	14	11
4096 KB	103	94	82	63	64	149	139	111	94	81
8192 KB	193	186	162	120	107	306	262	214	187	169
16384 KB	405	354	311	246	203	621	534	458	386	325
32768 KB	805	714	614	632	469	1234	1060	902	755	667

FIGURE 7: 256, 192, 128 BIT VS. 64/32 BIT ENCRYPTION RESULTS

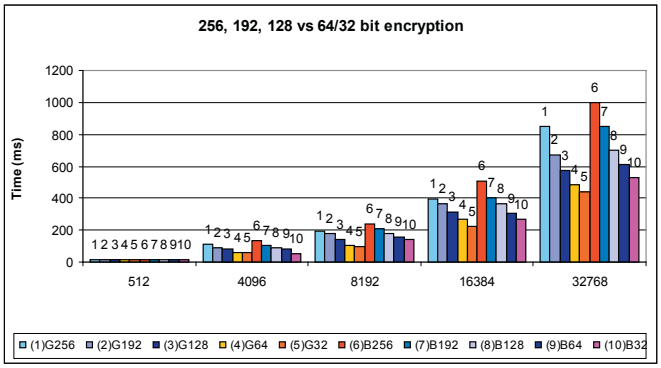
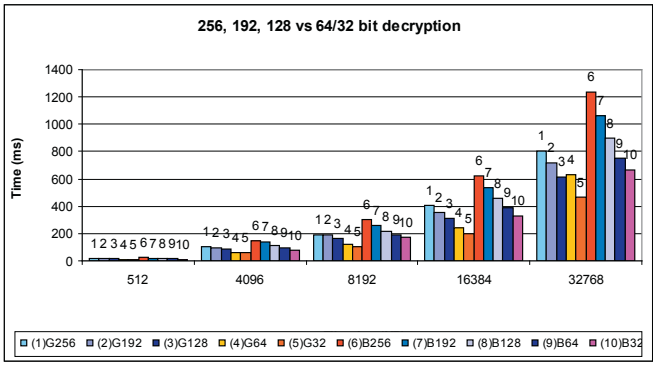


FIGURE 8: 128 BIT VS. 64/32 BIT DECRYPTION RESULTS



sults show that the time necessary for data processing is almost proportionally reduced as the number of algorithm rounds goes down.

CONCLUSION

Tested Oracle/SUN JCE and Bouncy Castle implementations use "provider based" architecture. According to our experimental results small, specialized implementations of the AES algorithm can be eqauly good or even faster than its large and multi-function counterparts. The multifunctional implementations take more time for initialization of proper algorithm or cryptographic tool, thus the data processing becomes longer.

Our tests have shown that when comparing well-known implementations, Bouncy Castle produces slightly preferable performances related to encryption time, while Oracle/Sun implementation is better when the criteria is decryption time. If we compare all implementations, EaesG brings equally good results as Bouncy Castle and Oracle/Sun when considering 128-bit encryption but slightly worse results when it comes to decryption. Both EaesG and BC appear to have equally preferable outcomes in the 192-bit encryption. However, taking into consideration the process of decryption, it is shown that Oracle/Sun implementation runs a bit faster. Finally, as we come to 256-bit encryption, all three implementations showed equally good results, while Oracle/Sun gets a better score in decryption.

Also, it should be mentioned that EaesG implementation based on Dr. Gladman's ideas shows significant improvements to EaesB implementation founded on Bertoni's idea no matter if it is related to encryption or decryption. On the other hand, it should be noted that EaesB implementation consumes significantly less memory, while still achieving satisfactory results.

We can point out that the conducted experiments have proven that AES algorithm can be expanded to 64 and 32 bit encryption given its high flexibility. This can lead to significant accelerations in its operation. Displayed results show that, depending on the number of both rounds and implementations, we can gain as much as 20-30% higher speed compared to 128 bit encryption and decryption.

From the presented experimental results it is clear that a certain acceleration can be achieved by constructing small and specialized implementation of AES algorithm instead of the use of the large implementations of the well-known software manufacturers. But the greatest advantage of constructing our own implementations is the possibility of further experimentation with a given algorithm for the purpose of research and comprehensive analysis.

ACKNOWLEDGMENTS

The work presented here was partially supported by the Serbian Ministry of Science and Technological development (project Multimodal biometry in identity management, contract no TR-32013).

REFERENCES

[1] Bertoni, G., et al. (2002). Efficient Software Implementation of AES on 32-Bit Platforms. CHES 2002: 159-171

[2] Carlos, C., et al. (2006). Algebraic Aspects of the Advanced Encryption Standard, Springer Science-Business Media, LLC.

[3] Daemen J., Rijmen V., (2002). The Design of Rijndael, Springer-Verlag, Inc.

[4] Damjanović, B. (2008), Implementation and extension of AES algorithm, Master's thesis, Faculty of Organizational Sciences, University of Belgrade,

[5] Dobbertin, H., et al. (2005). Advanced Encryption Standard AES, 4th International Conference, Bonn, Germany, 2004, Springer-Verlag

[6] Federal Information Processing Standards Publication 197, (2001). Specification for the ADVANCED ENCRYPTION STANDARD (AES), Available at: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf (Accessed: December 2011)

[7] Francia, G., et al. (2007). An Empirical Study on the Performance of Java/.Net Cryptographic APIs, Information Security Journal: A Global Perspective, 16: 6, 344 - 354

[8] Georges, A., et al., (2007). Statistically Rigorous Java Performance Evaluation, Department of Electronics and Information Systems, Ghent University, Belgium

[9] Gladman, B. (2007). A Specification for Rijndael, the AES Algorithm, Available at: http://gladman.plushost.co.uk/oldsite/cryptography_technology/rijndael/aes.spec.v316.pdf (Accessed: December 2011)

[10] Hook D., (2005). Beginning Cryptography with Java, Wrox Press

[11] http://www.bouncycastle.org/ (Accessed: December 2011)

[12] http://www.gladman.me.uk/ (Accessed: December 2011)

[13] Java SE security, http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136007.html (Accessed: December 2011)

[14] Konheim, A. (2007). Computer security and cryptography, John Wiley & Sons

[15] Van Etten, D., (2009). Why Many Java Performance Tests are Wrong, Available at: http://java.dzone.com/articles/why-many-java-performance-test/ (Accessed: December 2011)

[16] Weiss, J., (2004). Java cryptography extensions: practical guide for programmers, Morgan Kaufmann

Submitted: December 13, 2011
Accepted: December 31, 2011

JITA 1(2011) 2:127-135

The Way of Students' Efficiency Improvement in Knowledge Acquisition and Transfer Knowledge Model in Clarolina CMS

Nevzudin Buzađija

Master of computer science, Mješovita srednja tehnička škola Travnik

e-mail: nevzudinb@bih.net.ba

Case study

UDC **37.018.43:004.651**

Summary: In this work, throughout the research which was organized in one high school in Bosnia and Herzegovina, it will be shown the influence of exercises on the final result in the e-learning environment at the final test done by students. The research was conducted from the subject informatics in the I, II and III grade. The type of the questions were of multiple choices, addition and accession. The aim was to see how much influence these online exercises have on the final outcome which is demonstrated through the final informatics test done by students and which is done in a classical way in classroom after the finished teaching materials that were planned according to high school rules. In the research, it was taken account of making all preconditions available for easy experiment conducting with regard to technical securing preconditions for students access to blended system of teaching. Concerning the recent experience, it is noticeable that youth like the use of IT and communication devices. In order to secure all necessary conditions, it was conducted the survey among students about having technical preconditions of online access to testing and about students knowledge of work principle in the Claroline LMS platform. The aim was to increase motivation of high school students with regard to the use of online materials, because in high schools of Bosnia and Herzegovina almost nothing is undertaken when it comes to the implementation of new IKT possibilities.

Key words: knowledge transfer, blended learning, Claroline, e-learning, exercises, motivation and web technology

Introduction

At informatics teaching, usually it is used computers for practical realization of curriculum depending on school having informatics cabinets. Curriculums are obsolete and don't follow changes that happen in achievements regarding information and communication technologies. Because of the actual situation, there is no unique education system and patchwork prevails, as teachers are left to themselves and they create teaching process according to their discretion.

The work's goal is to find and prove the needs for implementation of new achievements in gaining and acquiring knowledge by students in informatics teaching, especially by those who are under average level. It is a word about students who follow teaching process slower, but also about those who want to learn more. Today, it is necessary to develop students' conscious about the need for continuing practicing for the purpose of achieving results that are necessary in order to prepare students for the use of gained knowledge. So when it comes to this, after finishing school, different methods are used in order to motivate students.

According students' conscious, there is not enough interest with regarding learning something new and achieving exceptional results which will help in creating life way, especially when it is a word about newer types of knowledge in the subject informatics.

Because of that, we want to explore in this work how and in which way the exercises created in the Claroline LMS platform have influence on the final outcome concerning the results which students show at the final test. The aim is to show in which way we can affect students' motivation, and at the same time that students want to accept it. In this work, we will show how much online exercises have influence on the results that students get in order to approach systematically to this problematic.

PROBLEM DESCRIPTION AND LITERATURE REVIEW

Since the beginning of the 1990s, the use of Web-based courses has been increasing constantly in all types of education and a tendency towards discovering new educational learning methods has emerged. A US study shows that among most colleges and universities (institutions with 15000 or more registered students), more than 96% of them offer online courses [1]. During the autumn of 2005, the same study showed that about 3.2 million students were enrolled in at least one online course in the US, about one million more than in the previous year. These studies show that an educational technological revolution has resulted in the increase of e-learning usage.

The development and use of any e-learning program represents an individual, organizational, and social investment. Therefore, the efficiency of e-learning should be evaluated. Efficiency measurement can represent a useful tool for the substantial decisions used in the application of any e-learning strategies. [4].

Ivankova [5] stated that a unique learning and teaching experience using a combination of research methods, through IT, teachers, and students, enriches the system of collecting information in a synchronized learning environment and gives the student a greater role in terms of knowledge acquisition. A distributed learning environment through IT affects students who attend traditional classes. The results showed that synchronized support through rich media presentations enhances the students' satisfaction with online courses.

Mccown [7] argues that combined courses can make the best use of both ways of teaching, online and traditional. There are many advantages for teachers and students, including flexibility and the students' increased participation in the process of acquiring knowledge. However, achieving this result is time-consuming.

A research study compared the group F2F and an online group, subjecting them to master a matter under the same conditions. The exam held for both groups showed that the online group was more efficient, according to both the exam results and the students' own perceptions [2].

In a study presented by Koenig [6], three groups with different work methods were formed in a classroom, using online and video conferencing, with the aim of comparing performances. The study showed that distribution in a classroom is more effective than the technology online distribution, and a bit more effective than video conferencing. It is also given a modality that can be used in universities in terms of different ways of knowledge distribution through the proper use of these three ways of knowledge distribution.

Many researches deal with blended learning as one of the possible learning systems, on that is in a way both simple and complex. There is a significant intuitive concept of advantage integration of synchronous and asynchronous learning in teaching activities. At the same time, there is a significant complexity in its implementation in challenges of almost unlimited possibility of the composition and applicability to so many contexts.

Online knowledge distribution in the US has an intensive use and it has been shown to be getting on construction rationalization of new facilities. On the other hand, faculty spend less time on lectures, and more time on interaction with students. Students spend less time passively listening to lectures, and more time actively participating in the course, solving tasks. Through their research works, many authors believe that blended learning demonstrates effectiveness, which justifies its further use. Of course, there are those who deny this, due to insufficient readiness

of the course participants, to devote more time to proper designing of the same, taking care of pedagogical and social principles [8]. The previous research that deals with blended learning in terms of students' attitudes in general show that, if there is a well-designed model combined with traditional class, students had positive attitudes toward the efficacy of blended learning [3]. Knowledge fortification is best achieved through use and practice in concrete assignments. The main usage of knowledge is step-by-step through end of chapter exercises. Advanced usage of knowledge is done at the end of every teaching unit (several chapters together). A user gets assignments which have to be solved through real application. The aim of the assignment is either explained throughout words or a desired final result is presented, and it is required to be done by the user himself.

Estimation tests are formed in order to simulate real problems. In that way, results show the real influence that a user's knowledge will have on business performance. During a knowledge check, results are associated with learning objects, and these instruct students to review the teaching units for which they did not get good results.

Sample

In the research, students of electrical engineering profession participated, profession: electrical engineer of computer technique and automatic – classes I, II and III. The classes were mixed by gender structure, namely – to 34 schoolgirls and 60 schoolboys. Students were also mixed by address, namely – to 32 students from urban area and 62 students from countryside.

Exercises that were the addition to the experimental group were created in the Claroline administrative frontend and were in the form of multiple choice, accession and addition. These exercises are independent variable because they were not taken into consideration during evaluation, but they had the role of improvement and increase of students' motivation.

Methodology

Before the experiment, we conducted anonymous surveying by the questionnaire with which we tried to get into the matter in the experimental group about: possession of preconditions for e-learning usage, reasons of using a computer, thoughts about subject informatics and desires concerning the results enhancement from this subject.

Also, we were interested in the possibility whether using computers and Internet can improve students' knowledge level and increase students' motivation towards acquirement of informatics matter through synchronized access and through communication with teacher by mail.

After conducted survey, we introduced students with e-learning terms, which offers us the way of material access at the Claroline system, the way of testing and knowledge evaluation and other elements which this system contains. As most of students of this group didn't have *mail* which is necessary for communication teacher-students towards the better communicating and informing, they got instruction to create mail on their name. After that, we approached the users' registration on the Claroline system towards the prevention of other students' registration for better following and conduction of the experiment. Every student got specific instructions in written form, username and password. After the beginning of the experiment, students could approach the system for practicing. That was necessary, especially for students of the first grades, who don't have enough previous knowledge, in order to eliminate all possible problems during the testing.

At students' recommendation, we also created themes within a forum which students could use in the Claroline system, mutually communicate at the posted topic.

After the experiment conduction, we conducted survey's questionnaire among students from the experimental groups, in order to deduce the level of understanding teaching materials done with the help of computers and ITS. We wanted to gain knowledge about the effects of some used system components (questions clarity at the Claroline platform), about stage fright appearance and its experience through positive or negative experience, and about pleasure and results of studying in given conditions.

Throughout this way, we wanted to evaluate the effect of this manner on the enhancement of students' motivation for studying. We stated earlier that very small students' motivation for studying teaching materials is present nowadays in schools.

During this research, there were used the results taken from questioning e-learning effect on students' success and motivation. With this research, we want to see how big influence of the exercises conducted in e-learning frontend can have on the results which students get on the final test.

Students acceded to exercises according to online tests after processed teaching themes at the class in classroom. Questions are conceptualized in such way that they covered the matter which was covered by teachers till that moment. Questions were varied by type and content, and they were questions of multiple choice, addition and connection between terms.

During the questioning that lasted two months, students had three exercises according to the principle when they had the obligation to access e-learning exercises after one thematic unit is being done. At the end, they did the final test in classroom through classical way of doing the final test.

During results processing, software Statistic 8 was used.

STUDENTS ATTITUDES

The important question is whether there are students' interest, motivation and readiness to learn and discover something new in this way, with IKT intercession. In the survey we conducted on 94 students, we asked: "If u had a choice, what would u chose?" We gave three possible answers on that question:

- Traditional,
- E-learning,
- Combination of traditional education and e-learning.

The results of survey are such that 73 examinees (77.6%) accepted the answer – combination of traditional and e-learning. These results show that students are interested for changes and acceptance of this kind of education. It is encouraging that interest exists, and especially that it is the biggest for that kind of education which currently have the biggest success in the world. On the other side, the result is even bigger when we know that this kind of education is still at the beginning in high school education. This term is related to the combination of traditional education and e-learning where the best elements of both types of education are included.

After the end of the research, the result of the survey is such that 79% of students completely understood the content they had in the Claroline system environment. 14% of them understood partially, and 6% of them didn't understand the content. Even most of them understood teaching matter, this shows that there are also those who didn't understand the matter partially or in no way, and that students should be more educated in order to use this kind of studying without problems.

The aim of the survey's question: "What is the informatics teaching in blended system in relation to classical teaching?", was to show students' attitude towards blended system. The result of the survey's question is such that 80% of students answered that the informatics content is more interesting with using e-learning system and most of students want to continue this kind of education. 74% of them want that continuously, while just 15% of them want it occasionally.

Also, it must be taken into consideration that students and teachers from our country mostly didn't have the experience with e-learning, but still they recognize its advantages. Here, the resistance to changes even for this kind of education is still present because of unknowing the world's trends and thoughts that e-learning doesn't have the same status as traditional education. So, this is also the difficulty of its development.

These students' attitudes justify this research towards the obtained methods and direction in which blended education system in high schools of Bosnia and Herzegovina should be developed. On the other side, it is necessary to define the individual effect of some elements , which LMS platform contains, on the students' results - in this case, the influence of the exercises on the students' results in classical tests.

Results and Discussions

The aim of the results which will be showed is to define part of exercises results and the influence of the same on the final students' result on the final test. For researching the influence, it was used the canonical-correlation and regression analysis.

These results primarily need to show at which way we can integrate e-learning with classical way of education. According to conducted surveys at the end of the research, it is noticeable that students are pleased with the way of communication among participants of this process and that this way suits them concerning the establishment of the matter. Also, it is noticeable that students understand questions in the context of answering, which makes them very interesting, especially because, after the finished testing in e-learning frontend, students points to mutual understanding of specific attitudes related to possible answers on the posted questions and to additional gaining knowledge about problematic that treats those questions. Students often use the forums created on the LMS platform as well as *chat* in mutual communication and communication with teacher. The advantage is that they get the results achieved during exercises right a way and that they can have insight in their results achieved in online exercises immediately.

Difficulties in constructing online exercises by students can appear in the case when we have loaded telecommunication networks, and also the time barrier which is there in the sense of answering on the posted questions. That is the case when it is wanted to prevent manipulation and abuse, so their time for creation online exercises is limited. So it happens that they don't have enough time for thinking. The second problem occurs concerning the acquirement of gained knowledge at this way, as practical uses of dealing a problem, which can occur in real practice, are not available for them.

The students' results of the exercises are shown under the marks V1, V2 and V3, while the students' results of the final test are under the mark ZT. The final test contained the informatics matter which was previously covered and which was practicing throughout e-learning exercises. The matter was constructed for all classes, and it was planned to be covered during that period according to informatics curriculum.

In this research, the results achieved in the II and III grade will be shown.

In the second grade – Table 1., as it is noticeable, only one latent dimension, which the overall effect on the result of the final test explains with 90.18%., while the structure of isolated canonic factors of exercises and final test was given in the table of canonic factors. Considering the structure of isolated canonical factor, it is noticeable that the results of the exercises achieved in V3-2 have the biggest influence, i.e. exercises that preceded the final test.

Table 1. Characteristic roots and explained parts of collective variance

	Eigenvalue	% Total - variance	Cumulative - Eigenvalue	Cumulative - %
1	3,607222	90,18055	3,607222	90,1806
2	0,212101	5,30252	3,819323	95,4831
3	0,109776	2,74440	3,929099	98,2275
4	0,070901	1,77253	4,000000	100,0000

Table 2. Canonic factor structure

	F1
V1-2	-0,955000
V2-2	-0,915003
V3-2	-0,962820
ZT-2	-0,964854
Expl.Var	3,607222
Prp.Totl	0,901806

In the table 3., the mutual connection of students' results achieved at some exercises and the results achieved on the final test can be noticed. It is also noticeable that there is a correlation between results of some exercises as it is a word about the matter which is mutually connected, and it couldn't just be observed as unique thematic unit concerning the matter covered for this informatics class.

Correlations are more than usual, probably because examinees are aware of belonging to the experimental group and it cannot affect intelligence development. That is why the real variability and collective co-variability were enlarged on all tests because of that. This fact gives us the stronger security in the interpretation of correlations between intelligence and knowledge.

TABLE 3. THE MATRIX OF VARIABLES INTER-CORRELATIONS

	V1-2	V2-2	V3-2	ZT-2
V1-2	1,00			
V2-2	0,82	1,00		
V3-2	0,90	0,83	1,00	
ZT-2	0,90	0,83	0,93	1,00

The value of canonic correlation coefficient of Table 4. is 0.891, Chi-square=61,088 with three levels of freedom; p=0,000 is statistically significant. It means that there is statistically significant connection between exercises and the final test.

Canonic correlation analysis was applied with the purpose of making maximum connections, i.e. relations between two observed data groups. With the use of Bartlet Lambd's test and his testing with suitable h'-square test, it is confirmed that the results achieved at some exercises are connected in some way to the results achieved on the final test, with one pair of canonic factor in the statistically significant level p=0.00. The connection between the first pairs of canonic factor is very large what confirms the size of canonic correlation coefficient which is R=0.944 and the explained part of collective variance from 89.1%.

TABLE 4. ISOLATED CANONIC FUNCTION

	Canonicl - R	Canonicl - R-sqr.	Chi-sqr.	df	p
0	0,944215	0,891542	61,08829	3	0,000000

Concerning the regression coefficients of BETA and its value Q (BETA), it can be concluded that the most influence on the final result will have those exercises which precede the testing, in this case (V2-3). But, the first exercise V2-1 also has statistically significant influence. Partial standardized regression coefficient for predictor variable of the first exercise is beta=0,326, t=2,154 with p=0,04, while for the third exercise is beta=0,552, t=3,612 with p=0,00.

Statistics show that the biggest partial effect on the results of the final test gives results at exercises which precede the test. While other exercise can have the smallest effect because questions in that test were with multiple choices only, so we can explain this decreased effect. Based on the values of non- standardized coefficients of regression, the regression equation can be formed in this way:

ZT-3=-16,22+0,379*V1-2+0,152*V2-2+0,832*V3-2

Based on this equation, the overall expected results can be expected on the final test.

TABLE 5. REGRESSION ANALYSIS

	Beta	Std.Err. - of Beta	B	Std.Err. - of B	t	p-level
Intercept			-16,2233	5,880102	-2,75902	0,010281
V1-2	0,325517	0,151107	0,3795	0,176170	2,15421	0,040310
V2-2	0,101868	0,120161	0,1518	0,179059	0,84776	0,404023
V3-2	0,552301	0,152922	0,8323	0,230439	3,61166	0,001224

In the III grade – Table 6., it can be seen that only one latent dimension was isolated. The latent dimension elaborates the overall effect on the results of the final test with 73.8%. The structure of isolated canonic factors of the exercises and final test is given. Considering the structure of isolated canonic factor, it is noticeable that the biggest effect have the results of the exercises achieved on V3-3, i.e. the exercises which preceded the final testing.

TABLE 6. CHARACTERISTIC ROOTS AND ELABORATED PARTS OF COLLECTIVE VARIANCE

	Eigenvalue	% Total - variance	Cumulative - Eigenvalue	Cumulative - %
1	2,953012	73,82531	2,953012	73,8253
2	0,549684	13,74210	3,502696	87,5674
3	0,353814	8,84535	3,856510	96,4128
4	0,143490	3,58724	4,000000	100,0000

TABLE 7. STRUCTURE OF CANONIC FACTOR

	F1
V1-3	-0,753953
V2-3	-0,833197
V3-3	-0,929705
ZT-3	-0,908845
Expl.Var	2,953012
Prp.Totl	0,738253

In the table of matrixes of variables inter-correlations, mutual effect of exercises and the final test can be noticed. This is similar as in other classes, although there is smaller intensity of correlation among results achieved during some exercises. This can be explained as in this class students study C++

language and it is a word about repeating orders which have their specificities. That correlation relationship tells us also about mutual similarities which control some of repeating orders.

Table 8. Matrix of inter-correlation variables

	V1-3	V2-3	V3-3	ZT-3
V1-3	1,00			
V2-3	0,48	1,00		
V3-3	0,58	0,73	1,00	
ZT-3	0,59	0,65	0,85	1,00

The value of canonic correlation coefficients is 0.733, Chi-square=40,343 with three levels of freedom, p=0.000 is statistically important. It means that there significant correlation between exercises and the final test.

Canonic correlation of analysis was applied for the purpose of defining maximal connectivity, i.e. relations between two monitored data groups. By applying Bartlet Lambd's test and his testing with the help of the suitable h' – square test, it was declared that there is, in a certain matter, the connection between the results achieved during exercises and the results of the final test with one pair of canonical factors at statistically envious level of p=0.00. The connectivity between the first pair of canonical factors is very high which is confirmed by the size of canonical correlation coefficient which is R=0'86 and the elaborated part of the collective variance of 73%.

Table 9. Isolated canonical function

	Canonicl - R	Canonicl – R-sqr.	Chi-sqr.	df	p
0	0,856498	0,733589	40,34282	3	0,000000

Based on regression coefficients of BETA and its importance Q (BETA), it can be concluded that exercises which precedes the testing will have the biggest influence on the final outcome in this case (V3-3). Partial standardized regression coefficient for predictor variable of the first exercise is beta=0,722, t=4,844 with p=0,00.

The data shows how the results during exercises which precede the test had the biggest partial influence on the final test results. On the other side, the other group of exercise had the smallest influence because questions in that test were only of multiple choice, so with that we can explain this small influence. Concerning the values of non-standardized regression coefficients, regression equation can be formed in this way:

$$ZT-3=-24,33+0,201V1-3+0,132V2-3+1,216V3-3$$

Based on this, we can calculate the overall results on the final test.

Table 10. Regression analysis

	Beta	Std.Err. - of Beta	B	Std.Err. - of B	t(30)	p-level
Intercept			-24,3309	10,61937	-2,29118	0,029150
V1-3	0,131781	0,116654	0,2013	0,17821	1,12967	0,267565
V2-3	0,067650	0,137465	0,1319	0,26802	0,49213	0,626212
V3-3	0,721873	0,149032	1,2159	0,25103	4,84375	0,000036

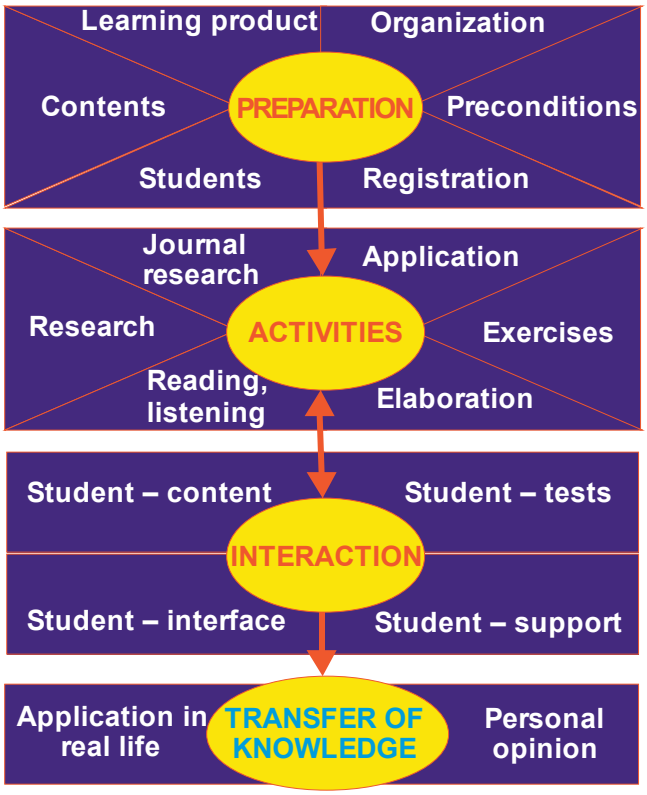
Throughout the results showed in this research, we can notice the influence of the predictor variables (exercises) on the criteria variable (the results achieved on the final test), concerning that this influence is more expressed in other classes. This research had multidimensional access towards checking the reliability of the data taken over this research. Of course, this initial researching is not completely reliable due to small results' number and short time period. In order to get the statistical reliable results, this research have to be followed in one longer time period and on the bigger samples' number.

But it is without a doubt that the influence exists, but we can't define precisely how big the real influence is. and through the results in practice of questioned population, the improvement is noticeable, not only through achieved results on the final test, but also the evident influence on the improvement related to students' motivation for discovering something new.

Knowledge Transfer

We further introduce another performance index, the knowledge transfer. The performance of knowledge transfer is closely associated with the

FIG.1. TRANSFER OF KNOWLEDGE



number of replicas of a given knowledge in the community. Different from the knowledge variety, the knowledge replica is defined as coexistence of homogeneous knowledge. In the knowledge sharing community, we need some identical knowledge to exist because this helps participants more easily gain certain type of knowledge from a "closer" community member. Since is the probability that the participant *i* shares certain knowledge, we denote the expected number of replicas of a type of knowledge in the community as $Ri(x1,x2,\dots,xn) = \sum_{i=1}^{n} x_i$ $Ri(x1,x2,\dots,xn) = \sum_{i=1}^{n} x_i$

We further assume that the transfer effort (e.g. delay) between any two participants is a random variable with value drawn from a transmission delay density function. Participants always retrieve knowledge from a community member with a minimum transfer effort. Denote the expected minimum transfer effort among *k* community members by T(k). Using order statistics, we have:

$$T(k) = \int_0^{\infty} t \cdot k \cdot (1 - F(t))^{k-1} \cdot f(t) \cdot dt$$

where *f*(*t*) and *F*(*t*) are the PDF and CDF for the transfer effort. In this paper, we analyze the community configuration based on uniform distribution $U[0,To]$ $U[0,To]$ where is the upper bound of transfer effort. Thus, given individual knowledge sharing level (x1, x2, ..., xn) , the expected transfer effort is:

$$T(R) = \left(\sum_{i=1}^{n} x_i + 1\right)^{-1} To$$

Lastly, we denote the value of a transfers knowledge as *vi* , and assume the cost of transfer effort and sharing cost for a knowledge are and respectively. The utility function is defined as follows:

$$U_i = v_i - \beta i T(R) - c_i x_i = v_i - \beta_i \left(\sum_{i=1}^{n} v_i + 1\right)^{-1} T_0 - c_i x_i$$

CONCLUSION

This research is in the context of the integration of contemporary discoveries and the IKT and Internet usage in the classical education system. Blended learning system is only possible in high schools, not only because of the legal barriers but also because of the social and pedagogical factor. Namely, in this kind of teaching, we don't lose the social and pedagogical principle because students still hang out and exchange their thoughts. On the other side, they have direct contact with teachers through classical education and in that way we get on quality of all factors needed to be pleased according students age towards the educational teaching as well as moral teaching.

Throughout the survey, with this kind of students' access, the pleasure is noticeable. There was needed even less time in order that students accept online teaching in comparison to teachers who have the negative attitude towards the same. It can be explained on the situation that in Bosnia and Herzegovina teachers don't have suitable informatics literacy so due to that, this resistance over blended learning system can be explained.

Throughout research, the increased influence of online exercises on the final test results is noticeable as the statistically significant influence can be noticed in the II and III grades. It is also noticeable in canonical-correlation and regression analysis which tested the achieved results. In both cases, the results matches which tells us about the reliability of the achieved results which were taken over the treatment

of exercises and final test of this population.

This research shows the universal solution for defining the influence of online exercises on the final test results which students achieve on the final test done in classical way. Qualitative changes were made after adequate conducted exercises where they took into account all aspects which were presented in experts research towards the creation of the same.

This research shows statistically significant influence of exercises on the final test results. It can be useful for the future research in this sector that would be conducted over longer period of time. Even the institutions which deal with creation of education strategy can benefit from it in order to strategically access the planning implementation for all online exercises subjects. However, besides all inaccuracy we mentioned, the survey also confirms that students have the interest towards this kind of testing, and in every case, the increased students' motivation is noticeable. Therefore, competent institutions should approach to the integration of this type of education due to students' results and the increase of informatics literacy of students and institutions' personnel.

References:

[1] Allen, E., & Seaman, J. (2006): Making The Grade: Online Education In The United States. Http://Www.Sloanc.Org/Publications/Survey/Pdf/Making The Grade. Pdf. (Active: 16 September, 2007)

[2] Carrol, N., Molly Burke (2010): Learning Effectiveness Using Different Teaching Modalities, American Journal of Business Education, 3, 12, ABI/INFORM Global pg 65

[3] Elizabeth, S., Philippa, G. (2007): Teaching For Blended Learning Research Perspectives From on Campus and Distance Students, Journal: Education and Information Technologies, Volume 12 Issue 3, September 2007, Kluwer Academic Publishers Hingham, MA, USA

[4] Figueira, E. (2003): Evaluating the Effectiveness of E-Learning Strategies For Small Aand Medium Enterprises, Available At: www.Theknownet. Com/ICT_Smes_Seminars/Papers/Figueira.Html (Active: 30 June, 2003)

[5] Ivankova, N. V. (2010): Teaching and Learning Mixed Methods Research in Computermediated Environment: Educational Gains and Challenges, Copyright © eContent Management Pty Ltd. International Journal of Multiple Research Approaches 4: 49–65

[6] Koenig, Robert J. (2010): A Study in Analyzing Effectiveness of Undergraduate Course Delivery: Classroom, Online and Video Conference From A Student and Faculty Perspective, Contemporary Issues in Education Research, 3, 10, ABI/INFORM Global pg.13

[7] Mccown, Linda J. (2010): Blended Courses: The Best of Online and Traditional Formats, Vol 23, No 4 Fall 2010 Clinical Laboratory Science 205

[8] Yukawa, J. (2010): Communities of Practice For Blended Learning: Toward an Integrated Model For Lis Education, Journal of Education for Library and Information Science, spring 2010, 51, 2, Research Library, pg.54

Submitted: December 13, 2011
Accepted: December 28, 2011

MONITORING OF JEE APPLICATIONS AND PERFORMANCE PREDICTION

Dušan Okanović, Milan Vidaković, Zora Konjović

{oki, minja, ftn_zora}@uns.ac.rs

Faculty of Technical Sciences, University of Novi Sad

Case study

UDC **005.334:004.4**

Abstract: *This paper presents one solution for continuous monitoring of JEE application. In order to reduce overhead, Kieker monitoring framework was used. This paper presents the architecture and basic functionality of the Kieker framework and how it can be extended for adaptive monitoring of JEE applications. Collected data was used for analysis of application performance. In order to predict application performance, regression analysis was employed.*

Key words: *continuous monitoring, Java, JMX, regression analysis*

INTRODUCTION

Degradation of software performance and quality of service over time is well known phenomenon [21]. Also, software testing, debugging and profiling in development phase are not able to detect everything that can happen after the software is deployed. New, previously unknown, errors can show up in this part of software lifecycle. It is necessary to monitor software over time in order to determine the software service levels i.e. how the software compares against service level agreements.

Although software developers usually use debuggers and profilers, there is often not enough time to properly test the software. Another problem with using profilers and debuggers is that they often induce an overhead, something the end user may find unacceptable. In order to determine how software behaves over time, in the real world, it is necessary to perform continuous monitoring of the software. The data provided by the continuous monitoring of software under production workload is much more valuable then the data obtained in the testing phase.

Monitoring system shares resources with the monitored software, causing the performance overhead. In order to control the overhead and the amount of data generated by the monitoring system, we can employ adaptive techniques. These techniques allow changing of monitoring parameters during monitoring process.

Obtained results can be used for visualization and performance analysis of software. Also, based on these results, we can predict how an application response time will change or when will some memory leak cause problem.

The main contribution of this paper is that it presents the use of open-source Kieker framework [20] with the extension for continuous monitoring of JEE applications. We created additional components that allow changing of monitoring parameters during monitoring process. By doing this, we can create flexible monitoring scenarios. As a case study, we present monitoring of a JEE application deployed on a cluster of servers. Results of this monitoring scenario are then used for application performance prediction.

In our earlier papers we presented some parts of this system. In [16] we proposed system's architecture, and in [15] we presented how this system can be applied for monitoring of applications deployed on the JBoss application server. Here, we show further improvements to the system and how the results we obtained can be used for performance analysis and prediction.

The remainder of this paper is structured as follows. Section 2 provides overview of related work in the field of performance monitoring and prediction. Section 3 presents architecture of our system, while section 4 shows its application to monitoring of one JEE test application. Performance prediction using linear regression is shown in section 5. Section 6 provides conclusion to this paper and guidelines for future work.

Related Work

Study presented in [19] indicates that performance is considered critical, but developers usually fail to use monitoring tools. In practice, application level monitoring tools, and especially open-source tools, are rarely used. The reasons for this are usually time constraints (during development), and resource constraints (e.g. performance degradation) during application use. Developers usually limit themselves to profilers and debuggers, during development.

Apart from Kieker, there are several other systems that are used for monitoring of distributed applications.

JBoss Profiler [9] is a tool based on JVMTI and JVMPI APIs. It is used to monitor applications deployed on JBoss application server [8]. The use of JVMTI/JVMPI APIs gives very precise results and low overhead. However, in order to change this tool or extend it, the knowledge of C/C# is required.

COMPAS JEEM [17] inserts software probes during the application startup. The probes are inserted into each of the layers (EJB, servlet…). The advantage of this approach is that there is no need for the application source code changes. However, a drawback of this approach is the fact that different probes must be defined for each application layer.

The system shown in [2] is used for reverse engineering of UML sequence diagrams from JEE applications. The instrumentation is performed using AspectJ, as is in Kieker. The system is limited to diagram generation and it is not suitable for monitoring. Also, the system is not able to monitor web-services, only RMI.

DynaTrace [2] and JXInsight [10] are examples of commercially available application monitoring tools. JXInsight is intended for JEE, while DynaTrace can be used for monitoring of .NET and Java applications. DynaTrace performs monitoring across multiple application tiers using PurePath technology. JXInsight is able to perform automatic analysis and detection of various problem types within applications.

One of the open-source tools that is often in use is Nagios [12], is not used on an application level, but to monitor infrastructure.

This overview shows the lack of tools (especially non-commercial open-source tools) that allow continuous and reconfigurable monitoring of JEE applications with low overhead. Kieker framework in combination with JMX [20] can be used for monitoring of JEE applications. It uses AspectJ [1] – load-time weaving configuration – for instrumentation and separation of monitoring code from application code. JMX, which is in the core of JEE application server infrastructure, can be used for controlling of the monitoring process.

Performance prediction of software is a part of capacity management process [18]. Developers usually use performance monitoring to obtain data for trend analysis. Prediction is also used in proactive management of software aging.

In [22] authors present their findings in the area of software aging and propose a proactive technique called "software rejuvenation". The idea is to occasionally terminate the application and clean its internal state of accumulated errors. This should be planned and initiated based on measurement, analysis and prediction.

Nudd et al. [13] provide a methodology for detailed performance prediction through software design and implementation cycles. It has relatively fast analysis time and can be used in runtime to assist in dynamically changing systems.

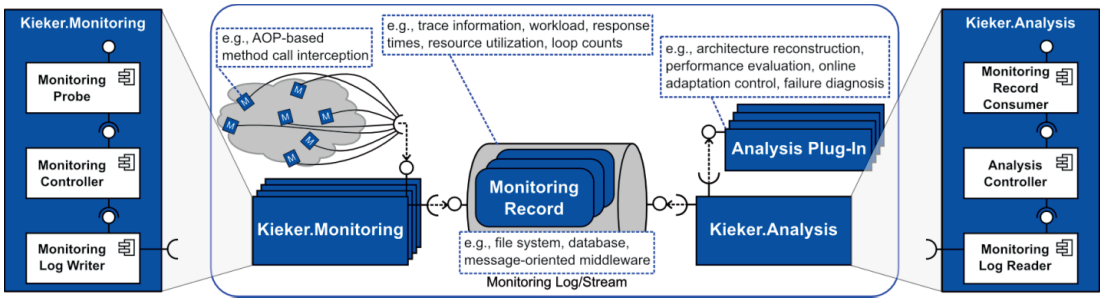
KIEKER FRAMEWORK

Kieker is a framework for continuous monitoring of all types of Java applications. It consists of:

- Kieker.Monitoring – component responsible for data collection and
- Kieker.Analysis – component that performs visualization of the data

Architecture of the Kieker framework is shown in fig. 1.

FIG. 1. KIEKER FRAMEWORK COMPONENT DIAGRAM



Kieker. Monitoring component is executed on the same computer where monitored application is being run. This component collects data on the execution of monitored applications. Monitoring Probe is a software probe that is inserted into the observed application and takes various measurements. Monitoring Log Writer stores collected data, in the form of MonitoringRecords, into the Monitoring Log. Monitoring Controller controls the work of this part of the framework.

The data in the Monitoring Log is analyzed by Kieker.Analysis component. Monitoring Log Reader reads records from Monitoring Log and forwards them to Analysis Plugin. Analysis Plugin analyzes and visualizes gathered data. Control of all components in this part of the Kieker framework is performed by Analysis Controller component.

Monitoring Log can be anything (e.g. file, database, JMS queue) because the framework does not depend on the type of storage.

Both components of the Kieker framework work completely independently. This approach allows a single computer to run monitored software, to store monitoring data in a file system or database on another computer and to perform data visualization and analysis on a third computer.

Software Instrumentation

Software instrumentation in the Kieker framework can be performed using aspect-oriented programming or by inserting pieces of code, which take measurements, create monitoring records and store these records using Kieker.Monitoring components. The drawback of the second approach is that it pollutes program code with the code that is not a part of the application. Use of aspect oriented programming is more appropriate way to perform program instrumentation. Developers can separate program logic from monitoring logic (separation of concerns). Instrumentation consists of writing aspect classes and weaving them with application classes. These aspects intercept execution of program logic at points defined using join points and add additional behavior using advices.

Among different AOP tools for the Java framework, Kieker framework uses AspectJ.

There are several ways to perform program instrumentation using AOP. Firstly, one can choose whether to instrument program code – i.e. weave aspects with application classes – during application development (compile-time weaving) or when classes are loaded (load-time weaving). Compile-time weaving is performed using AspetJ's *ajc* compiler: compiler weaves application code with aspects and generates new classes.

The other way to instrument the application is load time weaving. In this case, weaving of the precompiled aspects with application classes is performed during loading of classes. The disadvantage of this approach is that launching of applications takes a bit longer than in case of compile time weaving, but there is no need for source code and recompilation of the application. Load-time weaving configuration is performed with the aop.xml configuration file. In the aop.xml file we define aspects and parts of the software (classes, packages) that are to be woven together.

Developer can chose to monitor every method in every class or only designated ones. The usual way to designate methods and classes are Java annotations. `OperationExecutionMonitoringProbe` annotation and several different aspects are distributed with the Kieker framework and allow creation of different monitoring scenarios.

Regardless of the chosen scenario (compile or load time weaving, monitoring of all or only annotated methods), the aspect intercepts executed method, takes necessary measurements, lets the method execute, creates MonitoringRecord and, using Monitoring Controller, stores data into MonitoringLog. Within one application there can be multiple annotations and aspects, and they can perform various measurements.

Kieker Framework Extension

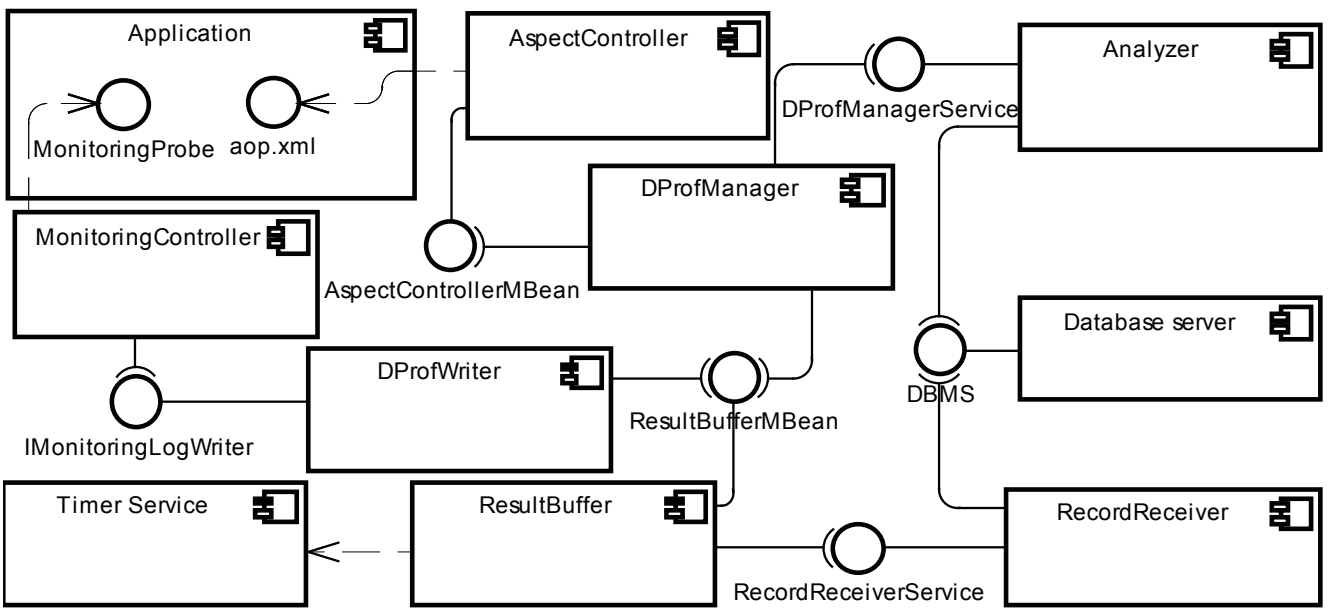
The Kieker framework was extended by implementing new MonitoringLogWriter and several new components. We call this new system the DProf.

Architecture of this part of the system is shown in fig. 2.

A new MonitoringLogWriter – DProfWriter stores all records into a special buffer – ResultBuffer. The ResultBuffer is implemented as a JMX MBean. This allows the buffer to be controlled programmatically or from any JMX console. The buffer sends monitoring records to a service running on a remote server – RecordReceiver. Records can be sent periodically in bulks or as soon as they arrive into the buffer. This remote service stores records into the database for further analysis. Essentially, the combination of the buffer, the service and database assumes the role of Kieker's Monitoring Log.

Analyzer component analyzes gathered data and sends new monitoring parameters to DProfManager. DProfManager controls ResultBuffer and AspectController. The configuration of monitoring system is performed through the aop.xml. AspectController performs monitoring system reconfiguration by adding and removing clauses from aop.xml.

Fig. 2. Extensions for Kieker framework



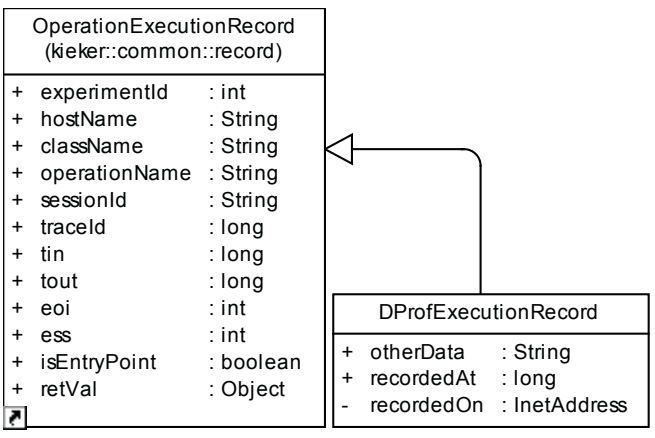
The system can be configured to:

- Record normal results – this is usually used to determine normal values of monitored parameters.
- Find which component is not in accordance with the expected values. In this case, the system monitors only top-level (interface) methods of components. If there is discrepancy with the expected values, the system turns on monitoring in the next level. The last method that has values different than expected is identified as the source of the problem.
- Find which component consumes selected resource the most. The process is similar to the previous. The difference is that there are no expected values. We only try to find on which level, which method consumes the most of the designated resource.

Another extension of the framework is addition of the new type of Monitoring Record – `DProfExecutionRecord`. It extends the standard Kieker's `OperationExecutionRecord` by adding new attributes. Attribute `recordedOn` holds the IP address of the computer where the record was created. Attribute `recordedAt` holds the time in milliseconds when the record was created. Because the original `OperationExecutionRecord` holds only information about response time, we have added the attribute `otherData`. It holds performance information of any other parameter, such as memory, CPU, network.

`OperationExecutionRecord` class is shown in Fig 3.

FIG. 3. OPERATIONEXECUTIONRECORD CLASS



CASE STUDY – FINDING PERFORMANCE BOTTLENECKS

The use of the Kieker framework for monitoring of distributed JEE applications will be demonstrated on the software configuration management (SCM) application described in [14] deployed on a JBoss 5.1.0 server. This is a JEE application responsible for tracking of applications and application versions.

The application is implemented using EJB technology. *Entity* EJBs [4] are used as O/R mapping layer. They are accessed through the *stateless session* EJB (SLSB), modeled on the *façade* design pattern [5]. SLSBs are annotated to work as JAX-WS web services as well.

Application client is the Java Swing [7] application which uses web services to access the application.

Listing 1. represents a part of the `OrganizationFacade` class. `createOrganization` method invokes `checkOrgName` method, retrieves object of `City` class by its id and creates a new entity EJB. All of these methods are annotated with `@OperationExecutionMonitoringProbe`.

Listing 1. Stateless session EJB `OrganizationFacade` class

```
@Stateless
public class OrgannizationFacade
     implements OrganizationFa-
cadeService {
     // ...
     @OperationExecutionMonitor-
ingProbe
     public Organization
createOrganization(String org-
Name,
     String address, String
email, long cityId) {
     checkOrgName(orgName);
     City c = entityManager.
find(City.class, cityId);
     Organization org =
     new Organization(orgName,
address, email, c);
     entityManager.persist(org);
```

```
        return org; }
        @OperationExecutionMonitor-
ingProbe
        public void checkOrgName() {
        // zip code check
        // ...
        }
}
```

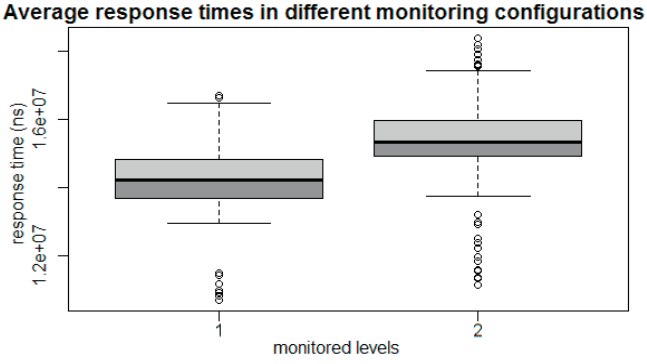
The testing will be conducted by repeatedly invoking `OrgannizationFacade. createOrganization (...)` method. These invocations are supposed to generate data which will be used for program performance analysis.

Initially, the system is configured for monitoring of methods in the top level – `createOrganization` method in this case. The system is configured to analyze monitoring data every two hours and change monitoring parameters, if needed.

In the first pass, results show that `createOrganization` method takes to long to execute. The monitoring system then included second level of methods into monitoring configuration. After two hours, the results were analyzed again. They have shown that average execution time of the `checkOrgName()` method is above expected. This method required refactoring, in order to meet demands.

Fig. 4. shows how response time changes when monitoring of another level is added to monitoring configuration.

FIG. 4. COMPARISON OF RESPONSE TIME WHEN ONE OR TWO LEVELS OF METHODS ARE MONITORED



We can see that the response time increases if another level of methods is added to monitoring configuration. By using adaptive monitoring technique, our system behaves as human tester would. It monitors only one level of methods, and turns on monitoring of lower level only if a problem is detected. This way, the total overhead is reduced.

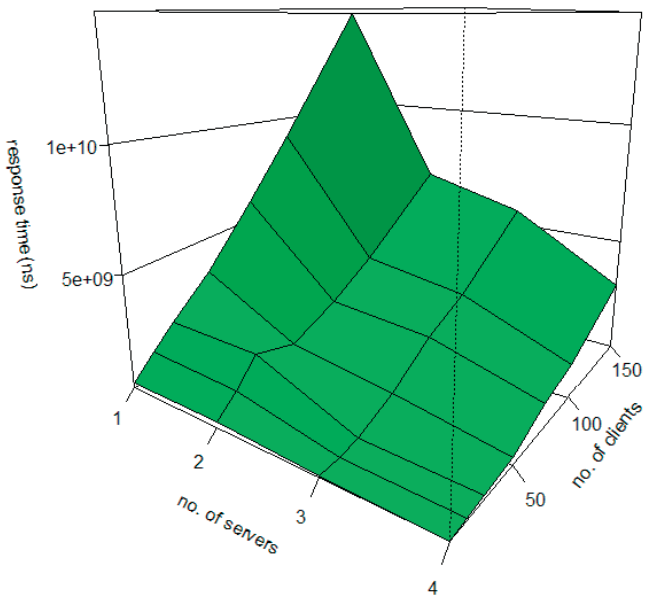
CASE STUDY – RESPONSE TIME PREDICTION

We have deployed our test application, from previous case, on a cluster of four servers and generated different workloads. We wanted to see what happens with the response time when we increase workload and number of servers in cluster.

Results are as expected: the response time increases with the number of clients and decreases with the number of servers.

Obtained results are shown in Fig. 5.

FIG. 5. AVERAGE RESPONSE TIME FOR DIFFERENT SERVER CONFIGURATIONS AND WORKLOADS



In order to predict how response time would change if the number of clients is increased or if we add another server, we employed the regression analysis. A model, in which we have two independent variables – the number of servers and the number of clients, and one dependent – response time, was created.

The analysis of this model shows that these two independent variables explain 83.4% of response

time variance. The rest of the average response time is influenced by some external factors. In this case, these can be hardware glitches, network traffic and cluster load balancer influence.

Both of the predictors are significant (the *p* value is less then 0.01) and the model provides us with coefficients for prediction shown in the table 1.

TABLE. 1. REGRESSION MODEL COEFFICIENTS

	Coefficients		Standardized Coefficients	t	Sig.
	B	Std. Error	Beta		
Intercept	$1.939 \cdot 10^9$	$4.965 \cdot 10^8$		3.906	0.000
Number of users	$6.221 \cdot 10^7$	4562536.524	0.820	13.636	0.000
Number of servers	$-1.254 \cdot 10^9$	$1.877 \cdot 10^8$	-0.402	-6.682	0.000

The following equation was derived from the table 1.:

$$\widetilde{RT} = 6.221 \cdot 10^7 \cdot N_{usr} - 1.254 \cdot 10^9 \cdot N_{ser} + 1.939 \cdot 10^9$$

($\widetilde{RT}\widetilde{RT}$ is estimated response time, $N_{usr}N_{usr}$ is number of users and $N_{ser}N_{ser}$ is number of servers). By using this equation, we can estimate (with the satisfying precision) how response time will change (with the respect to the calculated errors for every coefficient) if we vary the number of users and servers.

Regression results show that we can use this model for performance prediction with satisfactory precision.

CONCLUSION

This paper presents the use of the DProf system for continuous monitoring of distributed Java applications and the use of monitoring data for performance prediction.

It describes the Kieker framework, its architecture and configuration. The Kieker was used for monitoring of one SCM application which was implemented using EJB and web-services technologies. Additional components, implemented using JMX technology, allow for development of the reconfigurable application monitoring system. During the monitoring, it is possible to change monitoring parameters. The system can also be configured to change monitoring parameters automatically in order to provide more precise data or to reduce performance overhead.

We have applied the regression analysis in order to estimate application performance. The result was the model which allows us to predict what will happen to application performance if the number of clients changes or if we change the number of servers the application is deployed on.

Future work will focus on further improvements of monitoring system. Also we will try to apply other machine learning techniques in order to improve performance prediction model.

References

[1] AspectJ, http://www.eclipse.org/aspectj/

[2] Briand LC et al. (2006) Toward the reverse engineering of UML sequence diagrams for distributed Java software. *IEEE Transactions on Software Engineering*, 32(9), 642–663.

[3] Dynatrace, http://www.dynatrace.com/en/

[4] EJB 3.0, http://java.sun.com/products/ejb/

[5] Gamma E et al. (1994) Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Proffesional, Boston, USA.

[6] Grottke M et al. (2006) Analysis of Software Aging in a Web Server. *IEEE Transactions on Reliability*, 55(3), 411-420.

[7] Java Swing, http://java.sun.com/javase/6/docs/technotes/guides/swing

[8] JBoss Application Server, http://www.jboss.org/jbossas

[9] JBoss Profiler, www.jboss.org/jbossprofiler

[10] JXInsight, http://www.jinspired.com/products/jxinsight/

[11] Kiczales G. et al. (1997) Aspect-Oriented Programming. In Proceedings of ECOOP, pp. 313, Vienna, Austria

[12] Nagios, www.nagios.com

[13] Nudd GR et al. (2000) Pace-A Toolset for the Performance Prediction of Parallel and Distributed Systems. *International Journal of High Performance Computing Applications*, 14(3), 228-251.

[14] Okanović D and Vidaković M (2008) One Implementation of the System for Application Version Tracking and Automatic Updating. In Proceedings of the IASTED International Conference on Software Engineering 2008, pp 62–67**,** Innsbruck, Austria

[15] Okanović D and Vidaković M (2011) Performance Profiling of Java Enterprise Applications. In Proceedings of the International Conference on Internet Society Technology and Management, on CD, Kopaonik, Serbia,.

[16] Okanović D et al (2011) Towards Adaptive Monitoring of Java EE Applications. In Proceedings of the 5th International Conference on Information Technology, on CD, Amman, Jordan

[17] Parsons T et al. (2006) Non-Intrusive End-to-End Runtime Path Tracing for J2EE Systems. *IEEE Proceedings – Software*, 153(4), 149–161.

[18] Rudd C and Lloyd V (2007) Service Design. The Stationery Office, UK

[19] Snatzke RG (2008) Performance survey 2008. (available at http://www.codecentric.de/export/sites/www/resources/pdf/performance-survey-2008-web.pdf)

[20] Sullins BG and Whipple MB (2002) JMX in Action. Manning Publications, USA

[21] van Hoorn A et al. (2009) Continuous Monitoring of Software Services: Design and Application of the Kieker Framework. Technical report, Institut für Informatik, Oldenburg, 2009.

[22] Yilmaz C et al. (2005) Main Effects Screening: A Distributed Continuous Quality Assurance Process For Monitoring Performance Degradation in Evolving Software Systems. In Proceedings of the 27th International Conference on Software Engineering, pp 293-302, St. Louis, USA

Submitted: October 25, 2011
Accepted: December 31, 2011

INSTRUCTIONS FOR AUTHORS

The *Journal of Information Technology and Application (JITA)* publishes quality, original papers that contribute to the methodology of IT research as well as good examples of practical applications.

Authors are advised that adherence to the Instructions to Authors will help speed up the refereeing and production stages for most papers.

- Language and presentation
- Length of submissions
- Submission
- Contact details/biographies
- Title of the paper
- Abstract and keywords
- Figures and tables
- Sections
- Footnotes
- Special characters
- Spelling
- References
- Proofs
- PDF offprint
- Copyright and permissions
- Final material
- Correspondence
- Publication ethics

LANGUAGE AND PRESENTATION

Manuscripts should be written in English. All authors should obtain assistance in the editing of their papers for correct spelling and use of English grammar. Manuscripts should have double spacing, with ample margins and pages should be numbered consecutively. The Editors reserve the right to make changes that may clarify or condense papers where this is considered desirable.

LENGTH OF SUBMISSIONS

Papers should not normally exceed 12 Journal pages (about 8000 words). However, in certain circumstances (e.g., review papers) longer papers will be published.

SUBMISSION

Manuscripts must be submitted through the JITA online submission system.

Please read the instructions carefully before submitting your manuscript and ensure the main article files do not contain any author identifiable information.

Although PDF is acceptable for initial submission original source (i.e. MS Word) files will be required for typesetting etc.

CONTACT DETAILS/BIOGRAPHIES

A separate file containing the names and addresses of the authors, and the name and full contact details (full postal address, telephone, fax and e-mail) of the author to whom correspondence is to be directed should be uploaded at the time of submission (you should select Contact details/Biographies as the file type). This file is not shown to reviewers. This file should also contain short biographies for each author (50 words maximum each) which will appear at the end of their paper.

The authors' names and addresses must not appear in the body of the manuscript, to preserve anonymity. Manuscripts containing author details of any kind will be returned for correction.

TITLE OF THE PAPER

The title of the paper should not be longer than 16 words.

ABSTRACT AND KEYWORDS

The first page of the manuscript should contain a summary of not more than 200 words. This should be self-contained and understandable by the general reader outside the context of the full paper. You should also add 3 to 6 keywords.

Figures and tables

Figures which contain only textual rather than diagrammatic information should be designated Tables. Figures and tables should be numbered consecutively as they appear in the text. All figures and tables should have a caption.

Sections

Sections and subsections should be clearly differentiated but should not be numbered.

Footnotes

Papers must be written without the use of footnotes.

Special characters

Mathematical expressions and Greek or other symbols should be written clearly with ample spacing. Any unusual characters should be indicated on a separate sheet.

Spelling

Spelling must be consistent with the Concise Oxford Dictionary.

References

References in the text are indicated by the number in square brackets. If a referenced paper has three or more authors the reference should always appear as the first author followed by et al. References are listed alphabetically. All document types, both printed and electronic, are in the same list. References to the same author are listed chronologically, with the oldest on top. Journal titles should not be abbreviated.

Journal

Avramović ZŽ (1995) Method for evaluating the strength of retarding steps on a marshalling yard hump. *European Journal of Operational Research*, 85(1), 504–514.

Book

Walsham G (1993) Interpreting Information Systems in Organizations. Wiley, Chichester.

Contributed volume

Huberman AM and Miles MB (1994) Data Management and analysis methods. In Handbook of Qualitative Research (Denzin NK and Linkoln YS, Eds), pp 428-444, Sage, Thousand Oaks, California.

Conference Paper

Huberman AM and Miles MB (1994) Data Management and analysis methods. In Handbook of Qualitative Research (Denzin NK and Linkoln YS, Eds), pp 428-444, Sage, Thousand Oaks, California.

Unpublished reports/theses

Nandhakumar JJ (1993) The practice of executive information systems development: and in-depth case study. PhD Thesis, Department of Engineering, University of Cambridge.

Proofs

Proofs of papers will be sent to authors for checking. Alterations to diagrams should be avoided where possible. It will not be possible to accept major textual changes at this stage. Proofs must be returned to the publishers within 48 hours of receipt by fax, first-class post, airmail or courier. Failure to return the proof will result in the paper being delayed.

PDF offprint

Corresponding authors will receive a PDF of their article. This PDF offprint is provided for personal use. It is the responsibility of the corresponding author to pass the PDF offprint onto co-authors (if relevant) and ensure that they are aware of the conditions pertaining to its use.

The PDF must not be placed on a publicly-available website for general viewing, or otherwise distributed without seeking our permission, as this

would contravene our copyright policy and potentially damage the journal's circulation. Please visit **http://www.apeiron-journals.com/JITA/authors/rights_and_permissions.html** to see our latest copyright policy.

Copyright and permissions

The copyright of all material published in the Journal is held by Paneuropean University APEIRON. The author must complete and return the copyright form enclosed with the proofs.

Authors may submit papers which have been published elsewhere in a foreign language, provided permission has been obtained from the original publisher before submission.

Authors wishing to use material previously published in JITA should consult the publisher.

Final material

All final material must be submitted electronically in its original application format (MS Word is preferred). The file must correspond exactly to the final version of the manuscript.

Correspondence

Business correspondence and enquiries relating to advertising, subscriptions, back numbers or reprints should be addressed to the relevant person at:

Paneuropean University APEIRON
Journal JITA
Pere Krece 13, P.O.Box 51
78102 Banja Luka
Bosnia and Hercegovina / RS

Publication ethics

We take an active interest in issues and developments relating to publication ethics, such as plagiarism, falsification of data, fabrication of results and other areas of ethical misconduct. Please note that submitted manuscripts may be subject to checks using the corresponding service, in order to detect instances of overlapping and similar text.

JITA

PUBLISHER
Paneuropean University APEIRON,
Colledge of Information Technology
Banja Luka, Republic of Srpska, B&H
www.apeiron-uni.eu
Darko Uremović, Person Responsible for the Publisher
Aleksandra Vidović, Editor of University Publications

ISSN 2232-9625